Calhoun: The NPS Institutional Archive

DSpace Repository

Theses and Dissertations | 1. Thesis and Dissertation Collection, all items

1992-12

Mathematical modeling and control law development for the atmospheric monitoring and control system of the Controlled Environmental Research Chamber (CERC) at NASA Ames Research Center

Mathers, Bruce H.

Monterey, California. Naval Postgraduate School

http://hdl.handle.net/10945/23950

This publication is a work of the U.S. Government as defined in Title 17, United States Code, Section 101. Copyright protection is not available for this work in the United States.

Downloaded from NPS Archive: Calhoun

http://www.nps.edu/library

Calhoun is the Naval Postgraduate School's public access digital repository for research materials and institutional publications created by the NPS community. Calhoun is named for Professor of Mathematics Guy K. Calhoun, NPS's first appointed -- and published -- scholarly author.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

Approved for public release; distribution is unlimited.

Mathematical modeling and control law development
for the atmospheric monitoring and control system of the
Controlled Environmental Research Chamber (CERC)
at NASA Ames Research Center

by

Bruce H. Mathers
Lieutenant, United States Navy
B.S., University of Colorado

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN ASTRONAUTICAL ENGINEERING

from the

nclassified

ECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

a. REPORT SECURITY CLASSIFICATION NCLASSIFIED	1b RESTRICTIVE MARKINGS
a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.
b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
PERFORMING ORGANIZATION REPORT NUMBER(S)	S MONITORING ORGANIZATION REPORT NUMBER(S)
a. NAME OF PERFORMING ORGANIZATION aval Postgraduate School 6b. OFFICE SYMBOL (*If applicable*) 55	7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School
c. ADDRESS (*City, State, and ZIP Code*) Ionterey, CA 93943-5000	7b. ADDRESS (*City, State, and ZIP Code*) Monterey, CA 93943-5000
a. NAME OF FUNDING/SPONSORING RGANIZATION 8b. OFFICE SYMBOL (*If applicable*)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER
c. ADDRESS (*City, State, and ZIP Code*)	10. SOURCE OF FUNDING NUMBERS Program Element No / Project No / Task No. / Work Unit Accession Number

1. TITLE (*Include Security Classification*)

Iathematical Modeling and Control Law Development for the Atmospheric Monitoring and Control System of the Controlled Environment esearch Chamber at NASA Ames Research Center

2. PERSONAL AUTHOR(S) Mathers, Bruce H.

3a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (*year, month, day*)	15. PAGE COUNT
Iaster's Thesis	From To	December, 1992	127

6. SUPPLEMENTARY NOTATION

he views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. overnment.

7. COSATI CODES			18. SUBJECT TERMS (*continue on reverse if necessary and identify by block number*)
FIELD	GROUP	SUBGROUP	Control; Adaptive Control; Environmental Chambers; Hypobaric Chambers; Nonlinear Systems

9. ABSTRACT (*continue on reverse if necessary and identify by block number*)

he objective of this research is to develop a mathematical model and control algorithm for maintenance of the environmental system within the ontrolled Environmental Research Chamber (CERC) located at the National Aeronautics and Space Administration (NASA) Ames Research enter, Moffet Field, CA. The hypobaric research chamber is currently undergoing renovation as part of the Human Exploration Development roject (HEDP), an effort on behalf of NASA for advanced life support research. A broad overview of the chamber is provided which includes a hysical description, preliminary system hardware and associated performance, and potential experimental uses. A mathematical model of the hamber air mass is developed based on key energy and mass balances. Two methods of adaptive control have been implemented for the coupled ontrol of temperature, composition, pressure and humidity within the closed environment. Simulations testing the control algorithm erformance are conducted, including a step and modified ramp response. The results of the simulations indicate the adaptive methods performed ell for the model presented. Further research is required in refining the chamber model for algorithm optimization including integration of ardware dynamics.

0. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION	
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS REPORT ☐ DTIC USERS	Unclassified	
2a. NAME OF RESPONSIBLE INDIVIDUAL oberto Cristi	22b. TELEPHONE (*Include Area code*) (408) 646-2223	22c. OFFICE SYMBOL EC/cx

D FORM 1473, 84 MAR 83 APR edition may be used until exhausted
All other editions are obsolete

SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

ABSTRACT

The objective of this research is to develop a mathematical model and control algorithm for the maintenance of the environmental system within the Controlled Environment Research Chamber (CERC) located at the National Aeronautics and Space Administration (NASA) Ames Research Center, Moffet Field, CA. The hypobaric research chamber is currently undergoing renovation as part of the Human Exploration Development Project (HEDP), an effort on behalf of NASA for advanced life support research. A broad overview of the chamber is provided which includes a physical description, preliminary system hardware and associated performance, and potential experimental uses. A mathematical model of the chamber air mass has been developed based on key energy and mass balances. Two methods of adaptive control have been implemented for the coupled control of temperature, oxygen concentration, pressure and humidity within the closed environment. Simulations testing algorithm performance have been conducted, including a step and modified ramp response. The results of the simulations indicate the adaptive methods performed well for the model presented. Further research is required in refining the chamber model for algorithm optimization and validation including the integration of selected hardware dynamics.

M319
c.1

TABLE OF CONTENTS

I. INTRODUCTION 1

A. HUMAN EXPLORATION DEMONSTRATION PROJECT . . 1

B. CONTROLLED ENVIRONMENT RESEARCH CHAMBER 3

C. FUTURE EXPERIMENTS 6

D. PROBLEM STATEMENT 7

II. BACKGROUND 10

A. PSYCHROMETRICS 10

B. HYPOBARIC CHAMBER DESIGN 14

III. MATHEMATICAL MODELING 18

A. SYSTEMS OVERVIEW 18

1. Atmospheric Monitoring and Control Subsystem 18

2. Temperature and Humidity Control Subsystem 20

B. DYNAMIC EQUATIONS DEVELOPMENT 23

1. Assumptions 24

2. Nomenclature 26

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

3. State Variable Definition 28

a. Temperature 28

b. Pressure 33

c. Mass Fraction of Oxygen 34

d. Humidity 36

4. State Space Representation 37

IV. CONTROL LAW DEVELOPMENT 40

A. CLASSICAL FEEDBACK CONTROL 40

B. ADAPTIVE CONTROL 40

C. INDIRECT ADAPTIVE CONTROL 44

1. System Identification 44

2. Linear Quadratic Regulator 47

D. DIRECT ADAPTIVE CONTROL 50

1. System Identification 50

2. Control Algorithm 52

V. SIMULATION RESULTS 56

A. STRATEGY 56

1. Steady State Response 57

2. Step Response 60

a. Indirect Controller Performance 61

b. Direct Controller Performance 64

3. Load Disturbance 67

VI. CONCLUSIONS/RECOMMENDATIONS 75

A. SUMMARY 75

B. RECOMMENDATIONS 76

APPENDIX A. COMPUTER PROGRAMS 78

APPENDIX B. PARAMETRIC STUDIES 107

LIST OF REFERENCES 119

INITIAL DISTRIBUTION LIST 120

I. INTRODUCTION

A. HUMAN EXPLORATION DEMONSTRATION PROJECT

The exploration of new frontiers has always been a driving force in human evolution. A return to the lunar landscape and on to the planet Mars appears to be the next logical step beyond the establishment of the Space Station Freedom. The National Aeronautics and Space Administration (NASA) Ames Research Center, in support of the Space Exploration Initiative (SEI), has undertaken an internal, multi-division project referred to as the Human Exploration Demonstration Project (HEDP). This project has been portrayed as an avenue to address the advanced technology requirements necessary to establish an efficient, integrated human living environment on either a lunar or Martian planetary surface. Specifically, this high-tech environment will be capable of providing all the necessary life support functions in an efficient closed loop system. Included within the environment will be the extensive utilization of state of the art automation for both physiological and psychological monitoring, data acquisition, and overall habitat regulating systems. HEDP will provide a "test bed" for the research and development of these critical technologies.

Four basic goals have been identified for the Human Exploration Demonstration Project: [Ref. 1, p. 2].

- Provide a simulator for the demonstration and evalution of selected technologies in an integrated setting.
- Create a realistic environment for introduction of new technology.
- Enhance the technology development and evaluation process through synergistic cooperation of multiple Ames divisions.
- Identify promising technology concepts to programmatic Centers for new and existing NASA projects.

The spectrum and variety of potential experiments separates the HEDP project from similar individual development and demonstration efforts. Although the central theme will concentrate on manned space exploration and the technologies involved in closed habitats, the opportunities to develop systems encompassing the full space mission spectrum are numerous. The Human Exploration Demonstration Project, in an effort to meet all of the aforementioned requirements, will be comprised of four major facilities and/or dedicated systems. These sytems are the Controlled Environment Research Chamber (CERC), a lunar landscape physical simulation, robotic platforms and devices, and a human powered centrifuge. The majority of these facilities will be contained within existing buildings and/or entities currently available within the confines of the NASA Ames Research Center Complex, Moffet Field, CA.

B. CONTROLLED ENVIRONMENT RESEARCH CHAMBER

A critical component in the overall HEDP project is the Controlled Environment Research Chamber. Designed as a hypobaric facility for use in high altitude, human-based research in partial support of the NASA's drive to the moon, it apparently went into a state of disuse in 1975-76 [Ref. 1, p 5]. The existing facility is currently housed in Building N-239A. It consists of a main chamber vessel and an accompanying airlock (Figure 1.1). The primary chamber

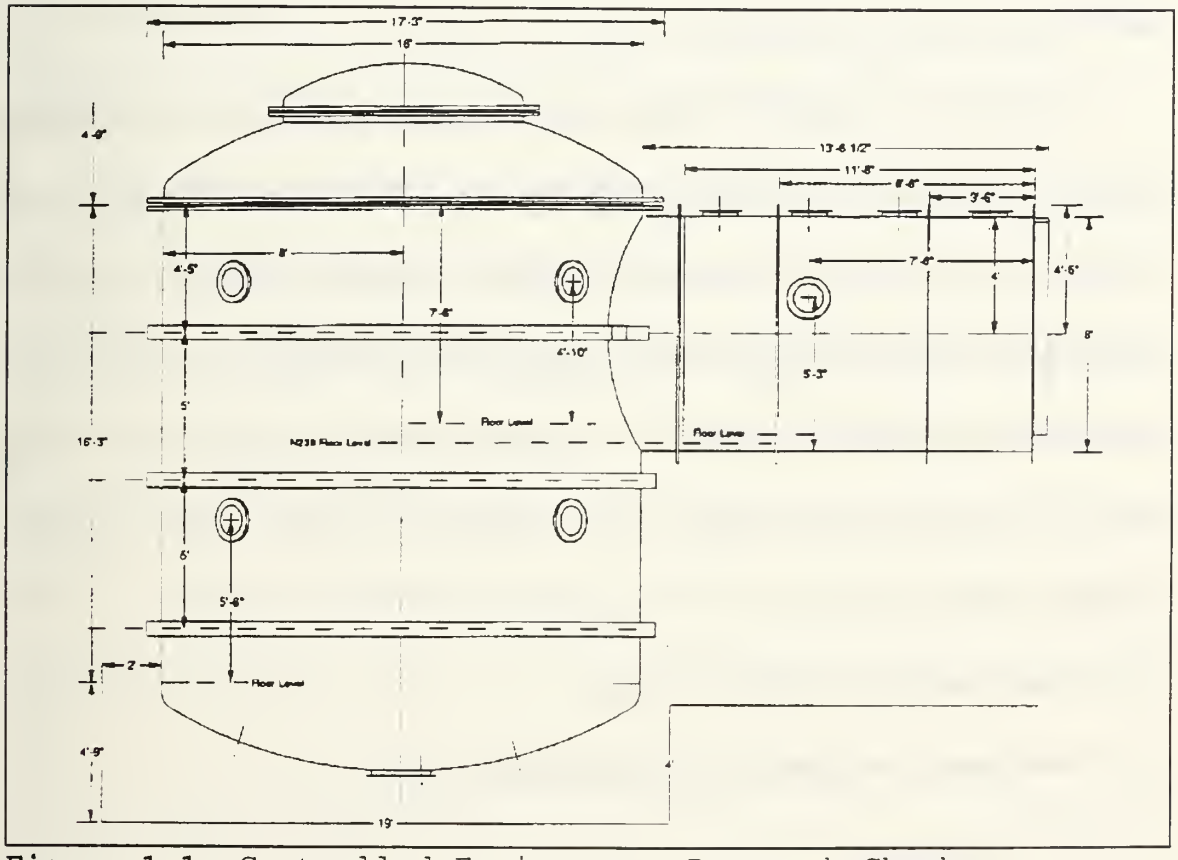


Figure 1.1 Controlled Environment Research Chamber

facility is a vertically oriented cylinder with an approximate diameter of 16 feet. The main body extends two levels in building N-239A and has an internal floor plan which consists of two separate deck spaces. The internal floorspace encompasses about 165 square feet. Movement between the levels is accomplished via a wall mounted ladder which extends to the second floor. The airlock was designed to provide a buffer for entrance into the main chamber. The airlock is a horizontal cylinder that is approximately 8 feet in diameter and provides air tight entries from both the outside into the airlock and from the airlock into the main chamber.

The habitat simulation will be housed within the Controlled Environment Research Chamber. The internal environment will provide a test bed for the evaluation of the various subsytems to sustain life support, verify algorithms and methods for the monitoring of crew health, and for the design of elements for the maintenance of personal hygeine. A typical lunar and/or Mars based habitat must be capable of providing certain minimum functions. Those functions include: [Ref. 1 p. 5].

- Temperature and humidity control
- Atmospheric monitoring and compositon control
- Air revitlization
- Water reclamation

- Solid Waste Management

The HEDP program definition currently only addresses the first three of the aforementioned functions. The temperature and humidity control susbsytem will basically function by either heating or cooling the constant ventilation flow through the chamber via a counter flow condensing heat exhanger. The humidity of the air will be controlled utilizing the same apparatus. The compostion of the air will be monitored and the oxygen-nitrogen levels of the environment will be tracked. Makeup of flows of oxygen and nitrogen will be provided via a supply of cryogenic gases. A system to regulate the removal and control of trace contaminants, including carbon dioxide, is also planned.

The inherent costs involved in the transport of materials into space demands design and implementation of these functions in the most proficient means achieveable. The testing and evaluation of the current life support technologies is best suited to a controlled environment that closely emulates the conditions under which the system would operate. The perilous environment found on both the lunar surface and the Martian landscape dictate tight, autonomous control of the habitat atmosphere system. The Controlled Environent Research Chamber will provide the avenue under which these functions and control algorithms may be demonstrated and operated.

C. FUTURE EXPERIMENTS

Future experiments for the Controlled Environment Research Chamber consist of four basic categories. The categories are:

- Loop closure maximization
- System integration issues
- Impact of spacecraft/habitat design issues on life support systems
- Bioregenerative or hybrid life support systems

Loop closure maximization will concentrate on the study of new and innovative life support process techologies involving air, water and waste reclamation and management. Experiments involving systems analysis and integration will revolve around the development and evaluation of sensors and control hardware, the variation in metabolic loads on life support overall system performance and integration, and on the long term logistic and reliability requirements of a closed life support system.

The study of the variation in environmental parameters on the life support system, to include the effects of pressure, composition, and humidity will emphasize another category of experiments. In addition, the monitoring of trace contaminants and quality of recycled water may be accomplished in this controlled environment. Finally, an examination of bioregenerative systems, which include the use of plants, may be accomplished. This will provide an

avenue for studies in plant growth optimization and determination of the life support system functions that may be supported by bioregenerative components.

D. PROBLEM STATEMENT

The goal of this thesis is to develop a mathematical model and control algorithm for the integrated control of the temperature, humidity, pressure, and composition of the habitat enclosed within the Controlled Environment Research Chamber. The model will include a basic review of hardware and sensors that have already been targeted for use in the chamber overhaul. A review of the expected oxygen and nitrogen consumption/makeup requirements in addition to an estimate of both the sensible and latent heat loads will be accomplished and a preliminary design to meet these requirements will be proposed. Concurrently, the aspects of oxygen deprivation and/or the bends phenomenon will be researched to ensure the control system is capable of safely supporting human inhabitants. Two different modern control methodologies will be examined to include an adaptive scheme of parameter estimation and control gain calculation and one which includes the use of an online parameter estimator coupled with a linear quadratic regulator control algorithm.

The system must be capable of providing a suitable environment through a varied set of operational conditions. In addition, the accuracy of control must

be maximized to the largest extent possible to fully utilize the chamber for a wide tightness of control requirements. A summary is provided in Table I.

Table I - CERC Operating Regime and Control Requirements

Parameter	Nominal Operating Range	Absolute Min/Max Values	Nominal Value	Control Accuracy Required
Pressure	34.464 - 101.325 kPa	34/102 kPa	Depends on Experimental Requirements	+/- 1% of Nominal Pressure
Temperature	18.33-26.67 °C	15.55/29.44 °C	21.1 °C	Depends on Experimental Requirements
Mass Fraction O2 (Dry Air)	0.64903-0.2321	0.649-0.232	Depends on Operational Pressure	+/- 2% of Nominal Pressure
Absolute Humidity	0.0079504 - 0.0232739 kg H_2O/kg dry air	25/70% Relative Humidity	Nominally Controlled @ 50% RH	Depends on Experimental Requirements

Critical to the overall control scheme is maintenance of the partial pressure of O2 when transitioning between experimental set points. Accuracy of control of the temperature and humidity will be nominally set at $\pm$ 1 °C and $\pm$ 5% of absolute humidity, respectively.

The response of the control system will then be analyzed with various distrubances and metabolic loads. The feasiblity of an adaptive control algorithm will then be reviewed for utilization in the future chamber design implementation.

II. BACKGROUND

The following sections detail two major subject areas requiring discussion with regards to the control of the environmental conditions within the Controlled Environment Research Chamber. Those areas include a brief synopsis on humidity and some insight into the impact of working in a reduced pressure environment.

A. PSYCHROMETRICS

Psychrometrics is the study of the impact of moisture on the properties of air. The simplest manner to obtain an inference into the moisture content of air is through an application of Dalton's law of partial pressures. This law simply states that in any nonreacting mixture of gases and vapors, each gaseous or vapor component exerts an individual partial pressure that is equal to the pressure that the gas would exert if it occupied the space alone, and that the total pressure is the sum of the mixture exerted by the individual gases or vapors [Ref 2, p. 21]. Hence, the total measured barometric pressure has a component representing the water vapor and is the basis for the definition of humidity.

It is important to recognize that the water vapor in air is actually steam at low pressure. The maximum amount of vapor that can be mixed with any given volume of dry air depends only on the temperature of the air. Since the amount of water vapor in the air determines the partial pressure exerted by the water vapor, it is evident that the air will contain the maximum amount of water vapor when the water vapor in the air exerts the maximum possible pressure. The maximum pressure that can be exerted by any vapor is known as the saturation pressure. The saturation pressure of water vapor at any given temperature may be determined utilizing the following relationship: [Ref 3, p. 59]

$$\log p = 28.5901 - 8.2\log(T+273) + 0.00248(T+273) - \frac{3142}{(T+273)} \qquad \textbf{(2.1)}$$

where T is the saturation temperature in °C and p is saturation pressure in bars.

The temperature corresponding to saturation is commonly referred to as the dew point temperature. The presence of 'dew' in the morning is an indication that the temperature dropped below the dew point temperature.

The water vapor content in the air is called humidity. The terms absolute and relative humidities are conventionally utilized to define the moisture state of the air. Relative humidity (ϕ) is the ratio of the mole fraction (or partial pressure) of water in moist air to the mole fraction (or partial pressure) of water

in saturated air at the same temperature and pressure. Relative humidity (RH) can be expressed functionally as

$$\phi = \frac{p_w}{p_s} \tag{2.2}$$

where p_w is the measured partial pressure of the water in the air

p_s is the saturation partial pressure of the water in the air

Absolute humidity, on the other hand, is defined at the actual density of water vapor in air. For many applications, it simplifies the analysis to define absolute humidity using a term known as the humidity ratio, w. This is simply the ratio of the mass of water to the mass of dry air. The humidity ratio at saturation may by calculated via the following relationship [Ref. 3, p.20].

$$w = \frac{M_w}{M_a} \frac{P_s}{P_{at} - P_s} \tag{2.3}$$

where M_w is the molecular weight of H_2O (kg/kg-mole)

M_a is the molecular weight of dry air (kg/kg-mole)

p_s is the partial pressure of H_2O in air (kPa)

p_a is the total atmospheric pressure of air (kPa)

Henceforth, any reference to absolute humidity will be actually a reference to the humidity ratio.

The methods of humidity control vary. The most common method of dehumidification is the utilization of an open coil condensing heat exchanger. Moist air is made to come in contact with a cooling coil, which is held at a temperature below the dew point of the air. The moist air is cooled and becomes saturated. The amount of moisture removed via condensation is a function of the mean coil temperature and the air dew point temperature. The difficulty encountered in dehumidification lies in the coupling of humidity and temperature. This is because a very small change in temperature can cause a very large change in humidity. The easiest manner in which to conceptualize the dehumidification

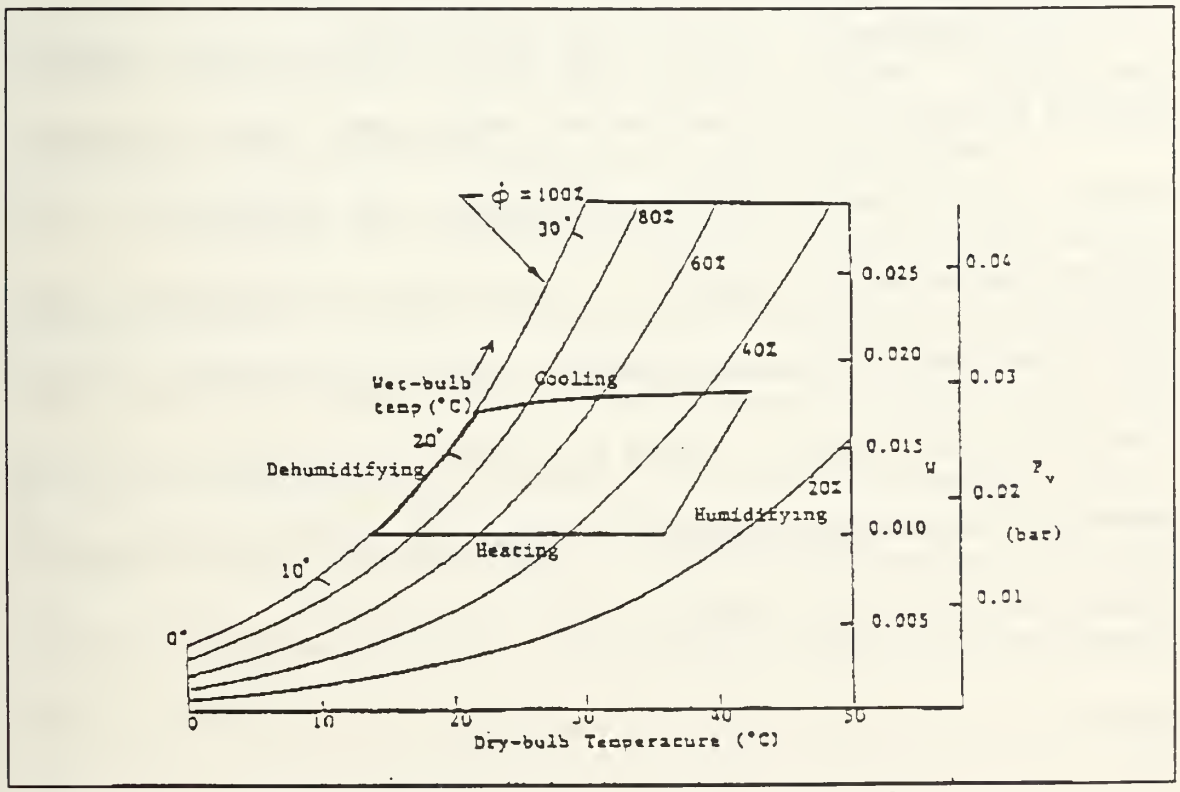


Figure 2.1 Psychrometric chart [Ref. 2, p.30]

process is to follow it on the psychrometric chart as depicted in Figure 2.1. All processes on the chart are at constant pressure. Starting at a point on the saturation (100% RH) curve, relative humidity varies from left to right. An increase in the sensible heat load is equivalent to a horizontal line in the same direction. It can be seen that starting from a point on the saturation curve, raising the temperature results in lower humidity. Moreover, it can be seen that the relationship between dry bulb temperature and relative humidity is exponential. This accounts for the fact that any change in temperature results in a much larger change in humidity.

B. HYPOBARIC CHAMBER DESIGN

Large decompression chambers and simulated habitats present a number of potential hazards to the personnel enclosed within them. From an atmospheric control point of view, one major consideration is the decompression effects derived from the improper transition to lower pressures and the impact of oxygen deprivation (hypoxia) or excess (hyperoxia). An artificial atmosphere of suitable composition and pressure is critical to the health and performance of crew members. This atmosphere supplies the oxygen their blood must absorb and the pressure their body fluids require.

Humans are accustomed to living in an environment that contains 21% oxygen by volume at 14.7 psia ambient pressure. According to Dr. R. Vann, Duke University, the optimum oxygen content is obtained when the partial pressure of oxygen is maintained at a nominal value of 160 mm Hg for all operating pressures below standard atmospheric pressure [Ref. 4]. Large variances from this value can cause physiological complications such as hypoxia or hyperoxia. Too little oxygen (hypoxia) induces sleeplessness, headaches, the inability to perform simple tasks, and eventually unconsciousness. Acute impairment of brain function occurs within 13 seconds whenever the aveolar oxygen tension drops below 33 mm Hg. Too much oxygen (hyperoxia) can lead to respiratory problems including inflammation of the lungs, heart disturbances, blindness, and loss of consciousness [Ref. 5, p. 5-1]. Figure 2.2 shows human performance limits versus total pressure and oxygen concentration.

In addition, the maintenance of the pressure and the proper timing of pressure transitions is crucial. There must exist a total pressure that prevents the vaporization of body fluids. This is referred to as an ebulism and occurs around one psia at a temperature of 37°C [Ref. 5, p. 5-4]. The transition from an atmospheric state to a hypobaric state in an improper or excessivley rapid manner can cause decompression sickness. Decompression sickness is the formation of small gas bubbles in the body tissues or vascular system and is commonly

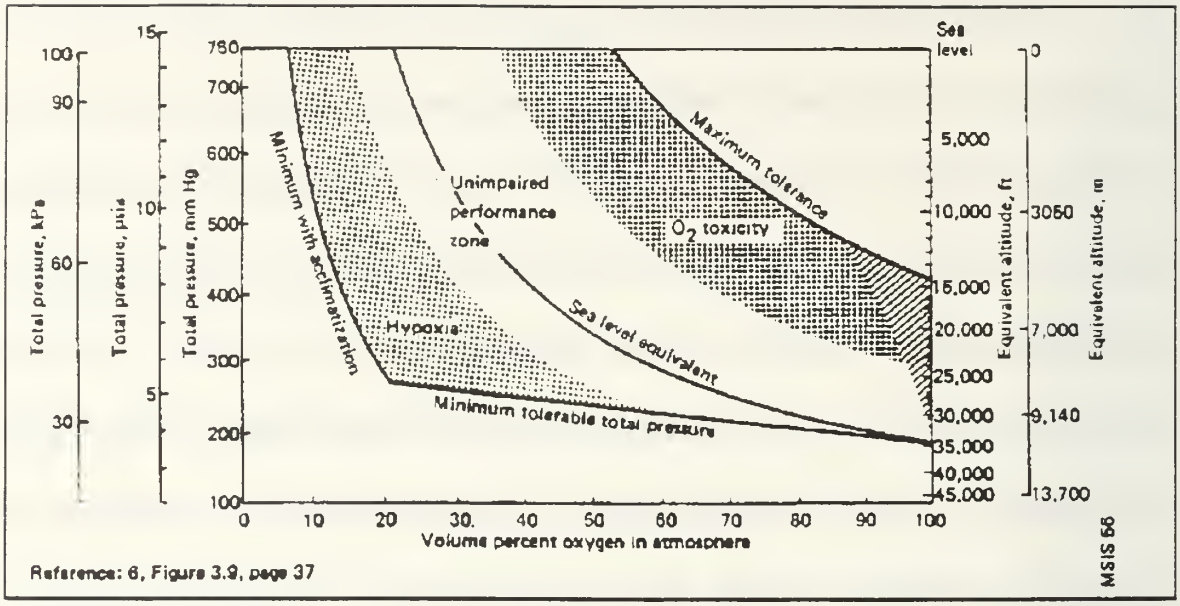


Figure 2.2 - Oxygen/Pressure Envelopes for Optimum Human Performance [Ref. 5, p. 5-1]

referred to as the bends. The seriousness of a case of the bends is related to an anatomic lodgement of one of these bubbles which may press on nerves or obstruct the blood supply. NASA currently utilizes a decompression scenario for Extra Vehicular Activities (EVA) in space suits during shuttle missions which requires a staged decompression. The pressure is reduced to around 10.2 psia where a period of pure oxygen pre-breath is conducted. The transition to the space suit pressure is then accomplished [Ref 6, p. 5].

For the likely scenarios involving the Controlled Environment Research Chamber, this was taken as the standard decompression scheme. A candidate control system must thus be capable of accurately meeting this

requirement while maintaining the oxygen partial pressure at a nominal value of 160 mm Hg.

From a controls point of view, it is imperative to accurately and properly transition from one pressure state to another. In addition, the concentration of oxygen must be maintained precisely to avoid either oxygen deprivation (hypoxia) or oxygen excess (hyperoxia). The wide range of operating pressures and varied experimental requirements for the Controlled Environment Research Chamber further complicates the control algorithm inception.

III. MATHEMATICAL MODELING

A. SYSTEMS OVERVIEW

The atmospheric control system is comprised of two distinct subsystems. One of the subsystems will provide the monitoring and control of the ambient pressure and concentration of oxygen. A cryogenic gas supply system will provide makeup gas with a vacuum line assisting in pressure control. This is known as the Atmospheric Monitoring and Control Subsystem. The temperature and humidity will be maintained via the Temperature and Humidity Control Subsystem. This system consists of temperature and humidity sensors coupled with a condensing heat exchanger capable of simultaneously removing excess moisture and maintaining chamber temperature.

1. Atmospheric Monitoring and Control Subsystem

The Atmospheric Monitoring and Control Subsystem is specifically tasked with the maintenance of the proper mix of atmoshperic gaseous constituents within the limits set for experimental requirements. In addition, the system is responsible for steady state pressure and transitional pressure control. The system is to provide autonomous operations utilizing a computer based

system with the capability to manually control the system in the event of an emergency condition.

The monitoring of the gaseous components will be accomplished via an Ohmeda Rascal II Anesthetic Gas Monitor. The Rascal II uses laser Raman scattering characteristics of gases to identify the gaseous constituents and relative concentrations. This device is capable of analyzing up to eight atmospheric gases with a minimum detectability of 0.25% by volume at a time response of less than or equal to 500 msec [Ref. 7, p.4]. Multiple sensors may be employed throughout the chamber and read through a multiplexer to obtain a more accurate overall view of the gaseous distribution. Concentrated pockets of a particular constituent may be detected utilizing this methodology and a potentially hazardous situation avoided. A pressure sensor has not been chosen for the chamber renovation.

Make up gases will be provided via a Linde cryogenic gas supply system. The gases will be mixed in a gas blending unit with the resulting mix injected into the chamber via the supply ducting in the ventilation system. Control of the various flows will be accomplished via a combination of a Linde flow meter console and Linde flow control modules. Each operator/flow metering console is capable of controlling the flow rates in four separate control modules. The operator console is configured with a RS232 connection for

external operation from a microprocessor or hands-off programmed operation by a central computing station. Manually operated valves will be provided for emergency operations. In addition, a completely separate oxygen supply system consisting of an independent piping system and gas masks available within the confines of the chamber will be provided as an emergency breathing alternative.

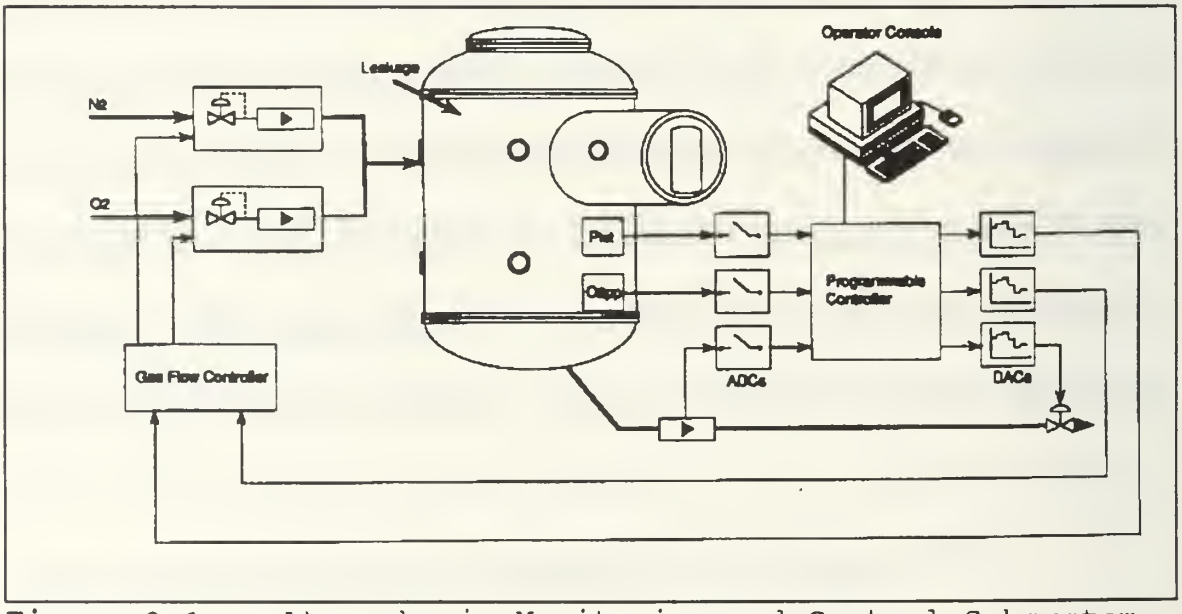


Figure 3.1 Atmospheric Monitoring and Control Subsystem

The basic system schematic is provided in Figure 3.1.

2. Temperature and Humidity Control Subsystem

The Temperature and Humidity Control Subsystem is designed to enhance human comfort by controlling the gas temperature and absolute humidity. It is also configured for autonomous operations with the capability to

be maunually overriden in the event of an emergency. A temperature sensor has not been chosen for the chamber overhaul process. For modeling purposes, a thermocouple has been selected for the chamber renovation. A thermocouple has an extremely fast response time ($<$ 1 sec) and requires no external power source. This fast response time will alleviate eventual modeling of a sensor time delay. The signal does, however, require some amplification and the output would require some additional conditioning due to nonlinearities in the output. For the tracking of humidity, a modern humidity sensor will be employed. Modern humidity sensors employ thin film technology and measure the ambient air dew point (from which the partial pressure may be derived) as a change in the capacitance of either aluminum hydroxide or silicon.

The ventilation system consists of an eight inch exhaust line and a twelve inch supply line. The internal ducting will be sized and configured to ensure that the air flow does not interfere with normal experimental procedures. The air mover is a sealed centrifugal fan which will have a nominal capacity of 3400 cubic feet per minute. The air flow will be channeled through air tight ducting into a sealed condensing heat exchanger. A damper mechanism will be utilized to control or divert the required flow over the condensing heat exchanger. The diverted flow will then be remixed with the heat exchanger flow prior to injection back into the chamber. The temperature will be primarily

controlled via the magnitude of air flow through the condensing heat exchanger. The humidity will be primarily controlled via coolant entrance temperature. Figure 3.2 provides a schematic of the Temperature and Humidity Control subsystem.

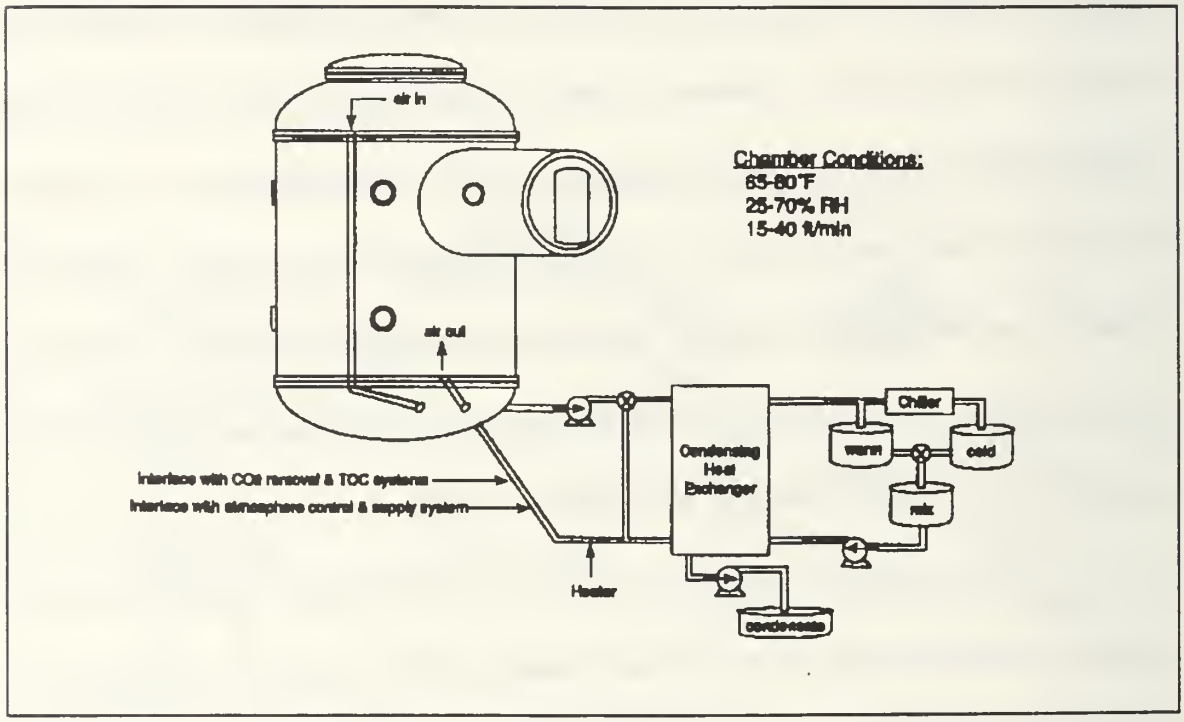


Figure 3.2 Temperature and Humidity Control Subsystem

A specific condensing heat exchanger has not been chosen for the chamber renovation project. However, parametric studies were conducted utilizing a conventional cross flow heat exchanger with a nominal heat transfer coefficient (U) of 10 btu/hr-ft^2-°F. Sensible and latent heat loads utilized in the studies were derived from information contained in reference 6 and will be

detailed in the following section. A 25% solution of calcium chloride brine was selected as the heat exchanger coolant because of its low freezing point (- 11°C) and non-toxic nature [ref. 9, p. 5]. The performance of the heat exchanger was defined in terms of nominal coolant rise temperature and median approach temperature. The nominal coolant rise temperature was modeled at 5.55 °C. Approach temperature, the difference between the temperature of the air that exits the heat exchanger and the coolant entrance temperature, was set at 8.33 °C. The parametric studies were conducted at various pressures between 34.4 and 101.325 kPa and the results are provided in Appendix B [Ref. 10]. The required heat exchanger was nominally sized at 5.6 m^2 of frontal area with a coolant flow rate of 178 kg/hr. These parameters were utilized in the mathematical modeling of the overall atmospheric control system.

B. DYNAMIC EQUATIONS DEVELOPMENT

The state of the air mass in the chamber at any given instant, given the assumptions provided in the following section, can be fully specified with four state variables. Those variables are pressure, mass fraction of oxygen in dry air, temperature and absolute humidity. The following subsections will detail the assumptions, nomenclature and equations utilized in the development of the chamber model representing the dynamics of these four variables.

1. Assumptions

The following is a listing of the major assumptions made in the development of the equations defining the air mass dynamics within the Closed Environment Research Chamber.

1. The gaseous consituents in the chamber are assumed to behave as ideal gases.

2. The chamber air is composed of only oxygen, nitrogen and water vapor. All contaminants and minor gases are excluded from consideration.

3. Potential energy of the chamber gas mass is assumed to be negligible.

4. The chamber is modeled as a lumped system to simplify analysis by avoiding partial derivatives.

5. The performance of the heat exchanger is fixed and is not varying over time.

6. The chamber is housed within a climate controlled building and hence the heat transfer through the walls into or out of the chamber is considered negligible.

7. The interior mean radiant and structure temperatures are assumed to be equal.

8. There are no objects in the chamber that absorb heat or moisture.

9. The velocity of the airflow through the chamber is assumed to be in a uniform direction and does not appreciably affect pressure (static pressure = stagnation pressure).

10. Ventilation flow through the chamber is assumed constant at 3400 cubic feet per minumte (cfm).

11. It is assumed that all makeup gases enter the chamber at ambient temperature and pressure and are perfectly mixed.

12. Particle concentrations, pressure, humidity and temperature are measured at the perfectly mixed conditions.

13. The heat capacities and thermal properties of the chamber constituents are assumed to be not a function of pressure.

14. The production of CO_2, derived through the metabolism of O_2, is assumed to be removed by the CO_2 removal subsytem as it is produced and hence is neglected.

15. It is assumed that there is no delay in the actuation of the controls and that the inputs have an instantaneous effect on the chamber states.

16. The maximum makeup flow rates of oxygen and nitrogen and vacuum are assumed to be 10 kg/hr.

17. The temperature range of coolant is assumed to be between 0 and 15 °C.

18. The chamber is assumed to have four human inhabitants with the following oxygen consumption and heat production loads. Three of the humans are assumed to be engaged in normal activity while one is involved in exercise/strenuous activity [Ref. 8, p. 10].

Table II SUMMARY OF CHAMBER LOADS

Source	Nominal O_2 consumption (kg/hr)	Sensible heat production (kJ/hr)	Latent heat (H_2O) production (kg/hr)
Three persons, normal activity	0.035	1326	0.285
One person, strenuous activity	0.349	664	0.455
Miscellaneous (Machinery, lighting, etc)	0.0	7720	0.0
Total	0.3848	9710	0.74

2. Nomenclature

The following is a listing of and definitions for the variables utilized in the system modeling.

$a_1, a_2, a_3, b_1, b_2, b_3$ = Coefficients of heat capacities for oxygen, nitrogen and water.

E_c = Total chamber air mass energy (kJ)

hvap = Heat of vaporization of moisture generated by humans (kJ/kg)

hvap1 = Heat of condensation of moisture in heat exchanger (kJ/kg)

H_v = Heat of vaporization at 0°C (kJ/kg)

H_2O = Total human latent load (moisture) (kg/hr)

m_{CHX} = Mass flow rate of air through heat exchanger (kg/hr)

m_{LEAK} = Leakage into the chamber (kg/hr)

m_{N2} = Mass flow rate of nitrogen into chamber (kg/hr)

m_{O2} = Mass flow rate of oxygen into chamber (kg/hr)

m_{OUT} = Mass flow rate of vacuum flow out of chamber (kg/hr)

M_a = Total chamber dry air mass (kg)

M_c = Total chamber air mass (kg)

M_{o2} = Total mass of oxygen (kg)

M_w = Total mass of moisture (kg)

O_2 = Total human consumption of oxygen (kg/hr)

P_c = Chamber Pressure (kPa)

P^* = Partial pressure of water at saturation (kPa)

q_m = Miscellaneous sensible loads (kJ/hr)

q_h = Total human sensible heat load (kJ/hr)

q_{CHX} = Total heat removed in heat exchanger (kJ/hr)

R = Universal gas constant 8.314 kJ/kg-mole- K

R' = Specific gas constant (kJ/kg - K)

T_c = Chamber temperature (°C)

T_e = Temperature of coolant entering heat exchanger (°C)

T_{exit} = Temperature of air exiting heat exchanger (°C)

u = Control variables

V_c = Chamber volume (m^3)

w_c = Chamber absolute humidity (kg H_2O/kg dry air)

x = State variables

x_c = Mass fraction of O_2 in dry air

$1-x_c$ = Mass fraction of N_2 in dry air

3. State Variable Definition

a. Temperature

Determination of the chamber bulk temperature involves an application of the generalized conservation of energy equation (open system formulation) in which the kinetic and potential energy terms are neglected.

$$\Sigma \dot{m}_i h_i = \Sigma \dot{m}_e h_e + \frac{dE}{dt} \tag{3.1}$$

The terms $\Sigma m_i h_i$ and $\Sigma m_e h_e$ represent system heat inputs and outputs, respectively, and E represents the total air mass energy. The total energy of the system is defined as the mass of air multiplied by its specific enthalpy. The mass of air may be derived utilizing the perfect gas law.

$$P_c V_c = nR(T_c + 273) \tag{3.2}$$

Because the humidity and mass fraction of oxygen are all defined in terms of dry air, the determination of the system energy will require referencing the system air mass to dry air. Rearranging the ideal gas law and inserting the specific gas constant R' for the unversal gas constant, the chamber dry air mass is defined as

$$M_c = \frac{P_c V_c}{R'(T_c + 273)} \frac{1}{1 + w_c} \tag{3.3}$$

where

$$R' = \frac{R}{(x_c(32) + (1 - x_c)(28) + w_c(18))} \tag{3.4}$$

The heat capacity of the overall air mass can be expressed utilizing the heat capacities of the individual gaseous components and multiplying by their respective mass fractions.

$$C_p air = x_c(a_1 + b_1 T_c) + (1 - x_c)(a_2 + b_2 Tc) + w_c(a_3 + b_3 T_c) \tag{3.5}$$

All of the above heat capacities are based on a specific temperature range with a minimum of 0°C [Ref. 2, p.496]. The enthalpy of the chamber air is defined as the sum of the heat capacities of the gaseous components multiplied by the chamber temperature and the heat of vaporization of water based. The heat of vaporization for this application will be referred to as H_v.

$$h_{air} = x_c(a_1 T_c + b_1 T_c^2) + (1 - x_c)(a_2 T_c + b_2 T_c^2) + w_c(H_v + a_3 T_c + b_3 T_c^2) \tag{3.6}$$

Combining equations 3.3 and 3.6, an expression for the total energy of the system may be obtained.

$$E_c = \frac{P_c V_c}{R'(T_c+273)} \frac{1}{1+w_c} [x_c(a_1 T_c + b_1 T_c^2) + (1-x_c)(a_2 T_c + b_2 T_c^2) + w_c(H_v + a_3 T_c + b_3 T_c^2)]$$

(3.7)

where H_v= 2213.35 kJ/kg [Ref. 1, p. 395].

Equation 3.7 must be differentiated with respect to each of the system variables in order to obtain an expression for dE_c/dt. Utilizing the chain rule, equation 3.7 was differentiated with respect to P_c, T_c, x_c and w_c. The result was then combined with the heat inputs and outputs into (3.1). For this analysis, the heat inputs are defined as

$$\Sigma h_i m_i = q_h + q_m + H_2O * hvap$$

(3.8)

and the heat outputs are defined as

$$\Sigma m_o h_o = q_{chx} \qquad \textbf{(3.9)}$$

The term q_{chx} is defined as the total heat (latent and sensible) removed from the chamber air mass via the condensing heat exchanger. As presented previously, the nominal exit temperature of the air leaving the heat exchanger has been modeled based on the approach temperature, which was assumed to be a measure

of performance for the condensing heat exchanger. Assuming the air exit temperature is a function of this approach temperature, the partial pressure of the moisture in the exit stream can be determined via equation (2.1) (assuming the air is saturated). The heat of vaporization (condensation) for the exit stream can then be calculated via an application of the Clausius-Clapeyron equation

$$hvap = (13.11 - 2.3 \log P^*) R' (T_e + 273) \qquad \textbf{(3.10)}$$

where P^* is in atmospheres and hvap is in kJ/kg [Ref 1, p. 15].

From equation (2.2), the exit humidity can be calculated. Incorporating this exit stream humidity into the following relationship provides the latent heat removed across the condensing heat exchanger.

$$q_l = \frac{m_{chx}}{1 + w_c} (w_c - w_e) hvap \qquad \textbf{(3.11)}$$

where w_e is the exit humidity. The sensible heat removed can be expressed as the difference of the heat capacities of the entrance and exit streams multiplied by their respective temperatures.

$$q_s = \frac{m_{chx}}{1 + w_c} [cp_{air-ent} T_c - cp_{air-exit} T_{exit}] \qquad \textbf{(3.12)}$$

where $cp_{air\text{-}ent}$ and $cp_{air\text{-}exit}$ represent the entrance and exit heat capacities, as defined previously. Incorporating these definitions into equation (3.1), an expression for the dynamics of temperature with respect to the other system variables may be obtained.

$$\frac{dT_c}{dt} = \left\{ \begin{array}{c} \left\{ \left\{ q_h + q_m + H_2O(hvap) - \frac{m_{chx}}{1+w_c} q_{chx} \right\} \frac{R'(1+w_c)(T_c+273)}{P_c V_c} \right\} \\ - \left\{ \left\{ -\frac{1}{1+w_c} \right\} (x_c(a_1 T_c + b_1 T_c^2) + (1-x_c)(a_2 T_c + b_2 T_c^2) + \right. \\ \left. (H_V + a_3 T_c + b_3 T_c^2) - \frac{w_c}{1+w_c}(H_v + a_3 T_c + b_3 T_c^2) \right\} \frac{dw_c}{dt} \\ -\left(a_1 T_c + b_1 T_c^2 - a_2 T_c - a_3 T_c^2\right) \frac{dx_c}{dt} \\ -\frac{1}{P_c}\left(x_c(a_1 T_c + b_1 T_c^2) + (1-x_c)(a_2 T_c + b_2 T_c^2) + w_c(H_v + a_3 T_c + b_3 T_c^2)\right) \frac{dP_c}{dt} \end{array} \right\} / Q \qquad \textbf{(3.13)}$$

where Q is defined as

$$Q=\left[\begin{array}{l} x_c\left[a_1-\frac{a_1T_c}{T_c+273}+2b_1T_c-\frac{b_1T_c^2}{T_c+273}\right]+(1-x_c)\left[a_2-\frac{a_2T_c}{T_c+273}+2b_2T_c-\frac{b_2T_c^2}{T_C+273}\right] \\ +w_c\left[-\frac{H_v}{T_c+273}+a_3-\frac{a_3T_c}{T_c+273}-\frac{b_3T_c^2}{T_c+273}\right] \end{array}\right] \tag{3.14}$$

b. Pressure

The pressure was modeled by invoking an overall mass balance on the chamber atmosphere. All mass consumption and production terms, including both the consumption and supply of oxygen, supply of nitrogen, air leakage into the chamber, removal through the vacuum line, and the moisture generation and removal terms were included. The generalized mass balance is defined as

$$\frac{dM_c}{dt}=\sum m_i-\sum m_o \tag{3.15}$$

where m_i and m_o represent the aforementioned mass inputs and outputs, respectively. Equation (3.14) can be expanded to include the respective chamber mass loads.

$$\frac{dM_c}{dt} = m_{O2} + m_{N2} + m_{LEAK} - m_{OUT} + H_2O - \frac{m_{chx}}{1 + w_c}(w_c - w_e) - O_2 \qquad \textbf{(3.16)}$$

An application of equation (3.2), the ideal gas law, provides a relationship for the overall chamber air mass.

$$M_c = \frac{P_c V_c}{R'(T_c + 273)} \qquad \textbf{(3.17)}$$

Differentiating equation (3.16), utilizing the chain rule, with respect to pressure and temperature, and solving for the differential of pressure with respect to time provides the equation of motion for pressure.

$$\frac{dP_c}{dt} = \frac{R'(T_c + 273)}{V_c}\left[m_{O2} + m_{N2} + m_{LEAK} + H_2O - \frac{m_{chx}}{1 + w_c}(w_{c_{w_e}}) - O_2\right] + \frac{P_c}{T_c + 273}\frac{dT_c}{dt} \qquad \textbf{(3.18)}$$

c. Mass Fraction of Oxygen

Determination of the dynamics of the chamber oxygen mass fraction involves another application of the generalized conservation of mass equation.

$$\frac{dM_{O2}}{dt} = \sum m_i - \sum m_o \tag{3.19}$$

The inputs and outputs, m_i and m_{out}, refer to the oxygen generation and consumption terms. The total oxygen in the air may be expressed as

$$M_{O2} = x_c M_a \tag{3.20}$$

where M_{O2} is the total mass of oxygen and M_a is the mass of dry air as defined in equation (3.3). Expanding equation (3.19) and inserting equation (3.20), the derivative of pressure becomes

$$d\left[\frac{x_c P_c V_c}{(1+w_c)(T_c+273)R'}\right] = (m_{O2} + x_l m_{LEAK} - x_c m_{OUT} - O_2) \tag{3.21}$$

Utilizing the chain rule and differentiating with respect to P_c, T_c, w_c and x_c, and solving for dx_c/dt.

$$\frac{dx_c}{dt} = \left[\begin{array}{l} \dfrac{(1+w_c)R'(T_c+273)}{V_c P_c}(m_{O2} + x_l m_{LEAK} - x_c m_{OUT} - O_2) - \dfrac{x_c}{P_c}\dfrac{dP_c}{dt} \\ \qquad + \dfrac{x_c}{(T_c+273)}\dfrac{dT_c}{dt} + \dfrac{x_c}{1+w_c}\dfrac{dw_c}{dt} \end{array}\right] \tag{3.22}$$

d. Humidity

The dynamics of the moisture can be determined via yet another application of a conservation of mass, this time on the moisture component.

$$\frac{dM_w}{dt} = \sum m_i - \sum m_o \tag{3.23}$$

where M_w is the total mass of the moisture in the air. M_w can be rewritten in terms of the humidity ratio and mass of dry air as

$$M_w = w_c M_a \tag{3.24}$$

The inputs into the system are the generation of moisture by humans, H_2O, and the output is the moisture removed by the condensing heat exchanger. These are defined as

$$\sum m_o = \frac{m_{chx}}{1 + w_c}(w_c - w_e) \tag{3.25}$$

$$\sum m_i = H_2O \tag{3.26}$$

Incorporating equation (3.3), which defines the mass of dry air, the differential of M_w with respect to time can be expanded

$$\frac{dM_w}{dt} = d\frac{\left[\frac{w_c P_c V_c}{(1+w_c)R'(T_c+273)}\right]}{dt} \tag{3.27}$$

Taking the derivative with respect to P_c, T_c, and w_c, and incorporating the expressions for the moisture inputs and outputs, the dynamics of the humidity can be expressed as

$$\frac{dw_c}{dt} = \frac{\left[\frac{R'(T_c+273)(1+w_c)}{P_c v_c w_c}\left[H_2O - \frac{(w_c - w_e)m_{chx}}{1+w_c}\right] - \frac{1}{P_c}\frac{dP_c}{dt} + \frac{1}{T_c+273}\frac{dT_c}{dt}\right]}{\left[\frac{1}{w_c} - \frac{1}{w_c+1}\right]} \tag{3.28}$$

4. State Space Representation

Assimilation of the equations representing the dynamics of pressure, mass fraction of oxygen, temperature and humidity into a simple set of equations was not feasible. The system is highly nonlinear, coupled and cannot be easily transposed into the standard state space form

$$\dot{x} = Ax + Bu + Cw \tag{3.29}$$

where x is the vector of state variables, u is the vector of control inputs, and w is the disturbance vector, which in this cases is the system loads. The state vector will be henceforth defined as follows

$$x = \begin{bmatrix} P_c \\ x_c \\ T_c \\ w_c \end{bmatrix} \tag{3.30}$$

and the control inputs as

$$u = \begin{bmatrix} m_{OUT} \\ m_{O2} \\ m_{N2} \\ T_e \\ m_{CHX} \end{bmatrix} \tag{3.31}$$

and the disturbance inputs, which are defined as H_2O, O_2,, q_l *and* q_s.

For simulation purposes, the system was separated into the following scheme

$$\dot{x} = F\ \dot{x} + G(x,u) + H(x,w) \tag{3.32}$$

The matrices F, G and H are presented in the computer code available in Appendix A. The matrix system was manipulated and solved for dx/dt in the following manner.

$$\dot{x} - F\dot{x} = G(x,u) + H(x,w) \tag{3.33}$$

$$\dot{x}(I - F) = G(x,u) + H(x,w) \tag{3.34}$$

$$\dot{x} = (I - F)^{-1}(G(x,u) + H(x,w)) \tag{3.35}$$

The system of equations were run to observe the dynamics of the modeled chamber. The results of the simulation were then analyzed to determine the maximum integration period required to ensure that the system dynamics would be accurately represented. The integration period that accomplished this goal was parametrically determined to be 15 seconds and a control sampling time of one and one-half minutes.

IV. CONTROL LAW DEVELOPMENT

A. CLASSICAL FEEDBACK CONTROL

The system of equations developed to model the chamber air mass are multi-variable and non-linear in nature. Classical control theory requires a multi-variable system to be decomposed into single-input/single-output (SISO) systems with separate transfer functions linking each combination of input and output signals. In other words, a system with three inputs and three outputs has nine transfer functions which must be arranged in matrix form. The poles of the multi-variable system are the poles of all the individual transfer functions. The poles of the overall system is the collection of poles of the individual subsystems. The zeros of the multi-variable system, however, do not correspond to the zeros of the individual transfer functions. The use of SISO techniques to design each individual subsystem with the expectation that the overall results will achieve the desired level of performance is unlikely. Hence, the use of modern control methods for this project has been pursued.

B. ADAPTIVE CONTROL

Linear control techniques are well defined and mature. Implementation of strict linear control practices on non-linear systems does not always give

satisfactory results. The response characteristics of non-linear processes utilizing linear control systems may vary significantly because they are normally linearized about one operating condition. The need to find a more sophisticated controller which could automatically adapt itself to the changing characteristics of the controlled process has lead to the development of adaptive control routines. For the application in this project, the term "adaptive control" represents one particular approach to designing a controller for the non-linear stochastic system. This approach consists of assuming that the controlled process satisfies linear equations about a particular operating condition only the values of the dynamic coefficients defining the equations are unknown. It is based on developing a routine which estimates these coefficients and utilizing the resulting estimates in the desired control law.

Although the system is nonlinear, it can be modeled using a linear state space representation about a nominal operating point. A parameter estimation scheme can then be used to identify the system parameters, assuming that the system can be considered linear around any given operating point. In a linear system, a controlled plant is described by the state-vector differential equation

$$\dot{x}=[A]x+[B]u \tag{4.1}$$

$$y=[C]x+[D]u \tag{4.2}$$

where

x	is the state variable vector
u	is the state input vector
y	is the state output vector
[A], [B], [C], [D]	are the coefficient matrices

The matrices A, B, C and D represent a model of the system. Key to obtaining an effective controller is developing or estimating the values for these matrices, which can be unknown or time varying, depending upon the particular operating condition. In an adaptive scheme, the components of the unknown matrices are estimated on-line by a suitable estimator. Based on the estimates of these parameters, a control scheme such as a linear quadratic regulator may be implemented for system control. This class of adaptive control is known as indirect adaptive control. Figure 4.1 provides a schematic of this method.

The direct mode of adaptive control is one which attempts to parameterize the unknown system directly in terms of the necessary control inputs in order to implement the desired control algorithm. For this mode, an estimator not only provides system identification but also predicts the control parameters and utilizes them directly in the controller. The basic schematic is provided in Figure 4.2.

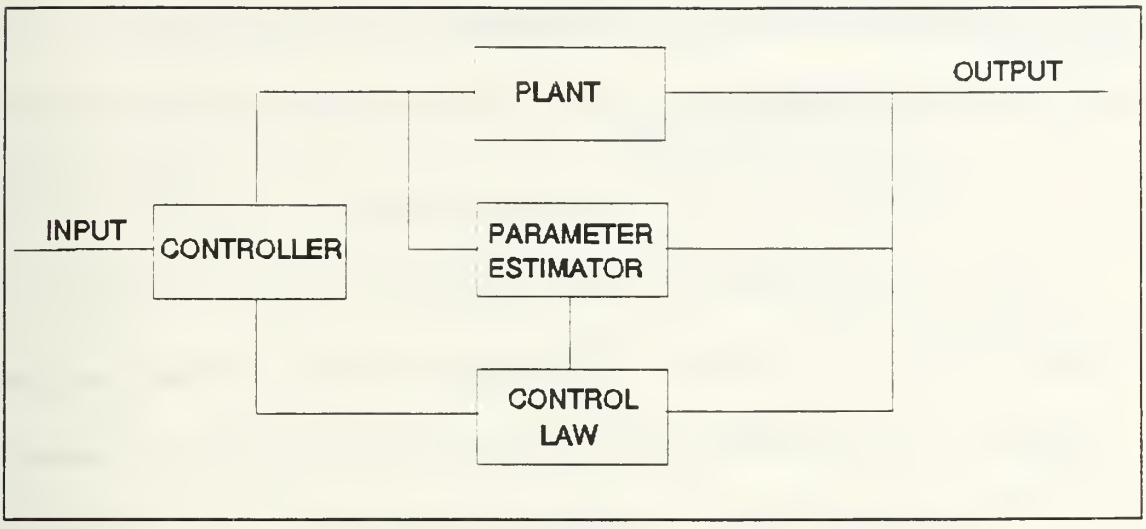


Figure 4.1. Indirect Adaptive Control

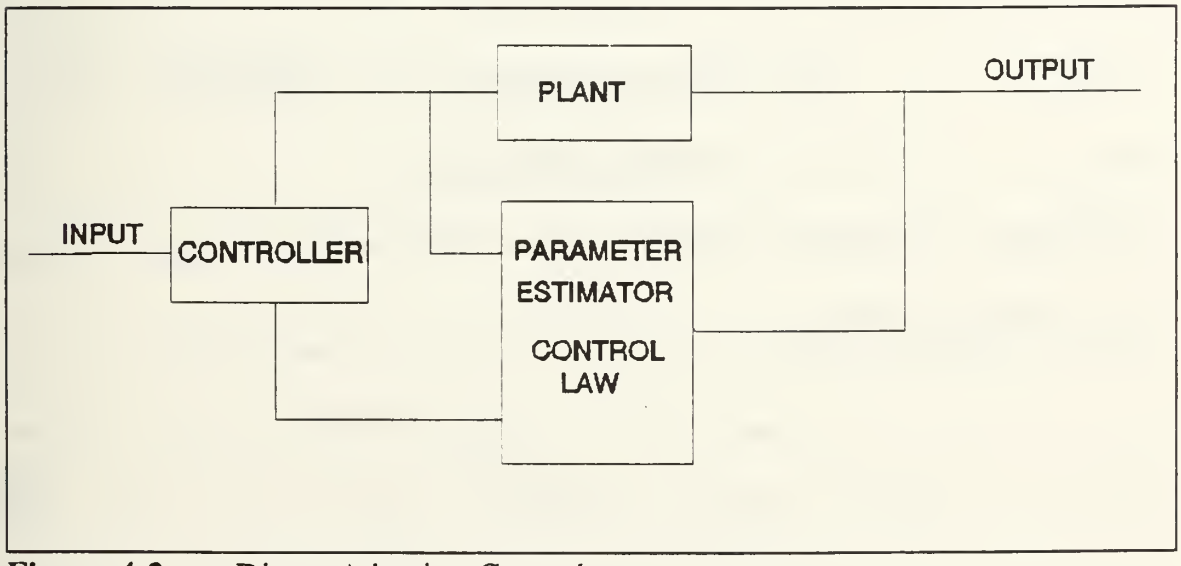


Figure 4.2 Direct Adaptive Control

C. INDIRECT ADAPTIVE CONTROL

1. System Identification

Parameter estimation/system identification is the process of deriving a model that best describes the system dynamics and closely matches the actual data presented. One method of determining this model is through least squares estimation. The principle of least squares states that the unknown parameters of a model can be determined by minimizing a square error criterion. This error is made of the sum of the squared differences between the acutal signals and those that are computed or estimated [Ref. 11, p. 420]. In the generalized least squares problem, it is assumed that the computed value takes the form

$$y=\phi^T(t)\theta \tag{4.3}$$

where y,ϕ are known signals (previous states/inputs)

θ are unknown parameters

The goal is to determine an estimate of the system parameter matrix θ such that the estimated variable

$$\hat{y}=\phi^T\hat{\theta} \tag{4.4}$$

is in close agreement with the actual signal, y. The least squares method states that the parameters should be chosen in a manner that minimizes the loss function [Ref. 12, p423]

$$J(\theta)=\frac{1}{2}\sum_{i=1}^{N}\epsilon_i^2 \tag{4.5}$$

where $\epsilon_i = y_i - \hat{y}_i = y_i - \theta_1\phi_1 - \ldots\ldots\ldots \theta_i\phi_i$

$i = 1, 2, \ldots N$ N = number of observations

The solution to this problem is given by the system of equations

$$\phi^T\phi\hat{\theta}=\phi^T y \tag{4.6}$$

If the matrix $\phi^T\phi$ is nonsingular, the minimum is unique and $\hat{\theta}$ can be determined by [Ref. 12, p. 422].

$$\hat{\theta}=(\phi^T\phi)^{-1}\phi^T y \tag{4.7}$$

For the situation in which observations are taken sequentially, as is the case with a discrete, sampled system, recursive equations may also be derived. This is known as a recursive least squares algorithm and leads to recursive identification. This algorithm takes the previous observations and the previous

least squares estimate for $\hat{\theta}$ to calculate the current estimate. The least squares estimate then satisfies the following set of recursive equations [Ref. 12, p.425].

$$\hat{\theta}_{k+1} = \hat{\theta}_k + P_{k+1}\left(y_{kt} - \phi_{kt}^T \hat{\theta}_k\right) \tag{4.8}$$

$$P_{k+1} = Q_{k+1}^{-1} \tag{4.9}$$

$$Q_{k+1} = Q_k + \phi_{kt}\phi_{kt}^T \tag{4.10}$$

where t is the sampling interval or period

k is the current sample i.e. k= 0, 1, 2, ...

and P is a matrix which is proportional to the variance of the estimates .

For the purposes of the chamber model, the following variables are defined.

$$z(t) = x(t) - x_0 \tag{4.11}$$

$$v(t) = u(t) - u_0 \tag{4.12}$$

where $x(t)$ is the vector of state variables

x_0 is the set point

$u(t)$ is the vector of control inputs

u_0 is the nominal control input at x_0

$z(t)$ and $v(t)$ are the deviations from the set points

Assuming the set point x_0 is fixed for a particular operating condition, the following relationship holds.

$$\dot{z}(t) = A_o z(t) + B_o v(t) \tag{4.13}$$

The measurable vector ϕ and calculated matrix θ are defined as

$$\phi = [z_1(t), z_2(t), z_3(t), z_4(t), v_1(t), v_2(t), v_3(t), v_4(t), v_5(t)]$$

$$\theta = \begin{bmatrix} a_{11} & a_{21} & a_{31} & a_{41} \\ a_{12} & a_{22} & a_{32} & a_{42} \\ a_{13} & a_{23} & a_{33} & a_{43} \\ a_{14} & a_{24} & a_{34} & a_{44} \\ b_{11} & b_{21} & b_{31} & b_{41} \\ b_{12} & b_{22} & b_{32} & b_{42} \\ b_{13} & b_{23} & b_{33} & b_{43} \\ b_{14} & b_{24} & b_{34} & b_{44} \\ b_{15} & b_{25} & b_{35} & b_{45} \end{bmatrix} \tag{4.14}$$

The recursive least squares algorithm is utilized online to estimate the coefficients in θ from measurements of $\dot{z}(t)$, $z(t)$, and $v(t)$. From this, A_o and B_o matrices from (4.11) are derived.

2. Linear Quadratic Regulator

Once we identify the system parameters by least squares, many of the linear, optimal control tools become available for use in control gain calculation. By optimal control theory we can determine the control input for the purpose of

maximizing a measure of performance or minimizing a cost function. If the system we want to control is uncertain, it is well known that performance has to be sacrificed for robustness. Performance is then measured by a maximization or minimization of a desired criterion. This leads to the concept of optimal stochastic control, a methodology which recognizes the random behavior of the system and attempts to optimize response or stability on the average rather than assured precision.

A discrete system is described by the state equation

$$x_{k+1}=A_k x_k+B_k u_k \tag{4.15}$$

The linear quadratic regulator algorithm is designed to find an optimal control $u^*(x(k), k)$ that minimizes the performance measure [Ref. 12, p. 78]

$$J=\frac{1}{2}\sum_{k=0}^{\infty}\left[x_k^T Q_k x_k+u_k^T R_k u_k\right] \tag{4.16}$$

where Q_k is a real symmetric positive semi-definite $n \times n$ matrix

R_k is a real symmetric positive definite $m \times m$ matrix

The minimization of the performance measure produces the control law in the form

$$u_k = -C_k x_k \tag{4.17}$$

where

$$C_k = R_k^{-1} B_k \lambda_{k+1} \tag{4.18}$$

and λ_k is an m-dimensional adjoint vector determined by

$$\lambda_k = Q_k x_k + A_k^T \lambda_{k+1} \tag{4.19}$$

For the indirect adaptive control algorithm, the matrix C_k is not a constant matrix but is changing over time in conformance with the new parameterization of the plant.

The critical aspect of minimizing the cost function is the proper choice in the weighting of the Q and R matrices. The larger the Q matrix in relation to the R matrix forces the minimization of the states (or the variance from set point) with less emphasis on the control effort. If R is proportionally larger than Q, the reverse is true. The control utilization is minimized at the expense of response time and state oscillations. The desired control system performance is thus driven by the selection of these parameters.

D. DIRECT ADAPTIVE CONTROL

Direct adaptive control, as specified earlier, combines the functions of parameter identification and control law calculation. For comparison and analysis purposes, the use of a direct adaptive control algorithm has been investigated. An existing, generalized direct adaptive control algorithm, written for the estimation and control of constrained multivariable systems, has been adapted and implemented for utilization in the chamber model. This generalized algorithm was written by Dr. Cory Finn, NASA Ames Research Center, and can be found in Reference 13. The following sections provide a brief discussion of the direct adaptive control algorithm.

1. System Identification

The state space representation utilized for implementation of the direct adaptive control system is referred to as the autoregressive moving-average (ARMA) model. This is an extension of the linear, discrete time state-space model

$$x(t+1) = Ax(t)+Bu(t) \tag{4.20}$$

$$y(t) = Cx(t) \tag{4.21}$$

which is rewritten as

$$A(q^{-1})y(t) = B(q^{-1})u(t-1) \tag{4.22}$$

where $A(q^{-1})$ and $B(q^{-1})$ are polynomials in the backward shift operator q^{-1}. These may be expanded as

$$A(q^{-1}) = 1 + a_1 q^{-1} + \ldots + a_n q^{-n} \tag{4.23}$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + \ldots + b_m q^{-m} \tag{4.24}$$

For the chamber model, the parameters of the polynomials $A(q^{-1})$ and $B(q^{-1})$ are unknown. These parameters may be estimated in terms of a system model.

$$\hat{A}(q^{-1})y(t) = \hat{B}(q^{-1})u(t-1) \tag{4.25}$$

The parameter estimates $\hat{A}(q^{-1})$ and $\hat{B}(q^{-1})$ are determined utilizing a recursive least squares algorithm similar to the one described in the previous section. For simplification, equation (4.25) is written as

$$\hat{y}(t) = \phi^T(t)\hat{\theta}(t) \tag{4.26}$$

where ϕ^T is a matrix of the past outputs and inputs and $\hat{\theta}$ is the parameter matrix. The parameter matrix is determined by using a recursive least squares estimation algorithm with a variable forgetting factor [Ref. 11]

$$\hat{\theta}(t) = \hat{\theta}(t-1) + \left[\frac{P(t-1)\phi(t-1)e(t)}{\lambda(t) + \phi^T(t-1)P(t-1)\phi(t-1)} \right] \tag{4.27}$$

$$P(t)=\frac{1}{\lambda(t)}\left[P(t-1)-\left[\frac{P(t-1)\phi(t-1)\phi^T(t-1)P(t-1)}{\lambda(t)+\phi^T(t-1)P(t-1)\phi(t-1)}\right]\right] \tag{4.28}$$

$$e(t)=y(t)-\phi^T(t-1)\hat{\theta}(t-1) \tag{4.29}$$

where $P(t)$ is the covariance matrix, and $e(t)$ is the prediction error.

The algorithm uses a variable forgetting factor which is defined as

$$\lambda(t)=\lambda(t-1)+\left[\frac{N_0}{1+\phi^T(t-1)P(t-1)\phi(t-1)}\right] \tag{4.30}$$

where N_0 is the memory length and is a constant [Ref. 14, p 831-835].

2. Control Algorithm

The generation of the necessary control inputs is accomplished via a constrained multivariable predictive controller [Ref. 13]. The predicted output signal at a time step t+k can be written as a linear function of previous inputs and output signals. Equation (4-26) then becomes

$$\hat{y}(t+k)=\hat{\alpha}(q^{-1})y(t)+\hat{\beta}(q^{-1})u(t+k-1) \tag{4.31}$$

where

$$\hat{\alpha}(q^{-1}) = \hat{a}_0 + \hat{\alpha}_1 + \ldots \hat{\alpha}_{n-1} q^{-(n-1)} \tag{4.32}$$

$$\hat{\beta}(q^{-1}) = \hat{\beta}_0 + \hat{\beta}_1 q^{-1} + \ldots + \hat{\beta}_{m+k-1} q^{-(m+k-1)} \tag{4.33}$$

The polynomials $\alpha(q^{-1})$ and $\beta(q^{-1})$ are obtained from the polynomials $\hat{A}(q^{-1})$ and $\hat{B}(q^{-1})$ by solving the Diophantine equations [Ref. 11, p.210].

$$1 = F(q^{-1})\hat{A}(q^{-1}) + q^{-k} G(q^{-1}) \tag{4.34}$$

$$F(q^{-1}) = 1 + f_1 q^{-1} + \ldots + f_{k-1} q^{-k+1} \tag{4.35}$$

$$G(q^{-1}) = g_0 + g_1 q^{-1} + \ldots + g_{n-1} q^{-n+1} \tag{4.36}$$

Given the values of $F(q^{-1})$ and $G(q^{-1})$, then

$$\hat{\alpha}(q^{-1}) = G(q^{-1}) \tag{4.37}$$

$$\hat{\beta}(q^{-1}) = F(q^{-1})\hat{B}(q^{-1}) \tag{4.38}$$

It is desired to solve the Diophantine equations for all values of k between one and the prediction horizon, T. The prediction horizon should be chosen such that it is larger than the process dead time, which is defined as the lag in system dynamic response to an input. This is necessary in order to obtain stable control.

Th goal is to solve the constained multivariable predictive control problem via an objective function, minimized with respect to the control input U

$$Obj=\min\left[w_1^T\,|\hat{Y}-Y^*|\;+\;w_2^T U\right] \tag{4.39}$$

subject to the constraints

$$\underline{U} \le U \le \overline{U} \tag{4.40}$$

$$-\delta \le \Delta U \le \delta \tag{4.41}$$

$$\underline{Y} \le \hat{Y} \le \overline{Y} \tag{4.42}$$

where Y^* is the vector of future set points

$\hat{Y}$ is the vector of predicted future inputs

U is the vector of future inputs

w_1^T is the vector of weights on the predicted future tracking errors

w_2^T is the vector of weights on the input efforts

$\underline{U}$ is the vector of lower bounds of the input efforts

$\overline{U}$ is the vector of upper bounds of the input efforts

δ is the vector of move size limitations on the manipulated input variables

$\underline{Y}$ is the vector of lower bounds on the output variables

$\overline{Y}$ is the vector of upper bounds on the output variables

For illustration purposes, $\hat{Y}$ and U are expanded as

$$\hat{Y}=\left[\hat{y}_1(t+1),\hat{y}_1(t+2),...,\hat{y}_1(t+T),\hat{y}_2(t+1),...,\hat{y}_2(t+T),...,\hat{y}_{kn}(t+1),...,\hat{y}_{kn}(t+T)\right] \tag{4.43}$$

$$U=\left[u_1(t),u_1(t+1),...,u_1(t+T-1),u_2(t),u_2(t+1),u_2(t+T-1),...,u_{km}(t),...,u_{km}(t+T-1)\right] \tag{4.44}$$

where kn represents the number of outputs in the process and km represents the number of inputs. For the chamber model, km and kn are equal to 4 and 5, respectively.

The above optimization can be cast as a linear programming problem, and in turn be solved using a simplex algorithm (Appendix A). The set of inputs, $u_1(t),u_2(t),\ldots u_{km}(t)$ calculated using the simplex algorithm are then implemented into the chamber for control. The future inputs, $u_1(t+1),\ldots,u_1(t+T-1),\ldots u_{km}(t+T-1)$ that are obtained from the optimization are never actually utilized for future control. This is because there is a new set of inputs are calculated at each time t for each time step.

V. SIMULATION RESULTS

A. STRATEGY

A critical first test of any preliminary control algorithm is to subject it to rigorous and realistic simulations. The model developed representing the dynamics of the chamber air mass provides the link for an initial performance study. The strategy for testing of the system includes a simulation of the steady state operational mode with minor pertubations in set point and a simulated transition or step response from a pressure of 34.464 kPa (5 psia) to 52.13 kPa (7.5 psia). The set point requirements for various points are provided in Table III. Finally, the control system is subjected to a series of incremental load increases.

Table III CHAMBER SET POINTS

Pressure (kPa)	O_2 Mass Fraction (Dry air)	N_2 Mass Fraction (Dry air)	Temperature (°C)	Absolute Humidity (kg H_2O/kg dry air)
101.325	0.231574	0.768426	21.1	0.0079504
86.16	0.272867	0.727133	21.1	0.0093431
68.92	0.338166	0.661834	21.1	0.0116668
51.69	0.444653	0.555347	21.1	0.015537
34.46	0.64903	0.35097	21.1	0.0232739

1. Steady State Response

In this section, we addressed the performance of only the author derived indirect adaptive controller/algorithm. A system simulation representing nominal steady state loads has been conducted. This loading is defined in Table II. The nominal control inputs necessary to maintain the set points were derived utilizing mass and energy balances over the entire system. Appendix B contains the results of the parametric studies utilized to obtain steady state values for the condensing heat exchanger performance. Simple mass balances were used to determine the necessary oxygen and nitrogen makeup flows in addition to the required vacuum flow.

The simulation is initiated by varying the control inputs randomly within a 10 percent range of the steady state values. This was accomplished for two reasons: (1) to allow the recursive least squares algorithm to "learn" the system model and allow the estimated parameter matrix $\hat{\theta}$ to converge, and (2) to allow the system to drift from the initial set point. Three hundred time steps, equivalent to 1.25 hours of simulation, has been alloted for the converging period. At this juncture, the controller was turned on. The dynamic response of the system is represented in Figures 5.1, 5.2, 5.3 and 5.4. When the control system is "turned on", the states fluctuate slightly but remain within the limits of control. The fluctuations that can be observed for the first hour in the plot in Figure 5.1 is

representative of the system "learning". The least squares estimator managed to recursively converge relatively quickly and presents an excellent model of the plant at this operating condition. The controller response to the minor set point pertubation and adaptation to the steady state scenario is satisfactory. However, a steady state error is observed in all states. The largest error encountered is in temperature. The temperature settled into a steady state value of 21.4°C, which translates into a steady state error of 1.4%.

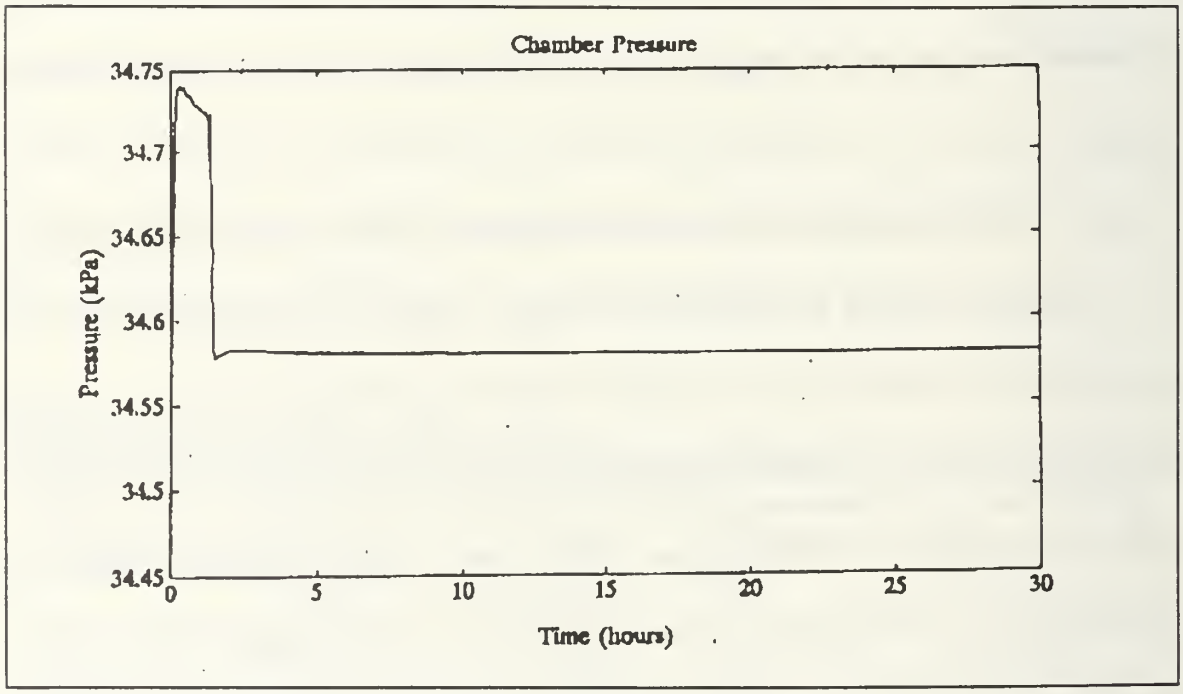


Figure 5.1 Pressure Steady State Response at 34.464 kPa

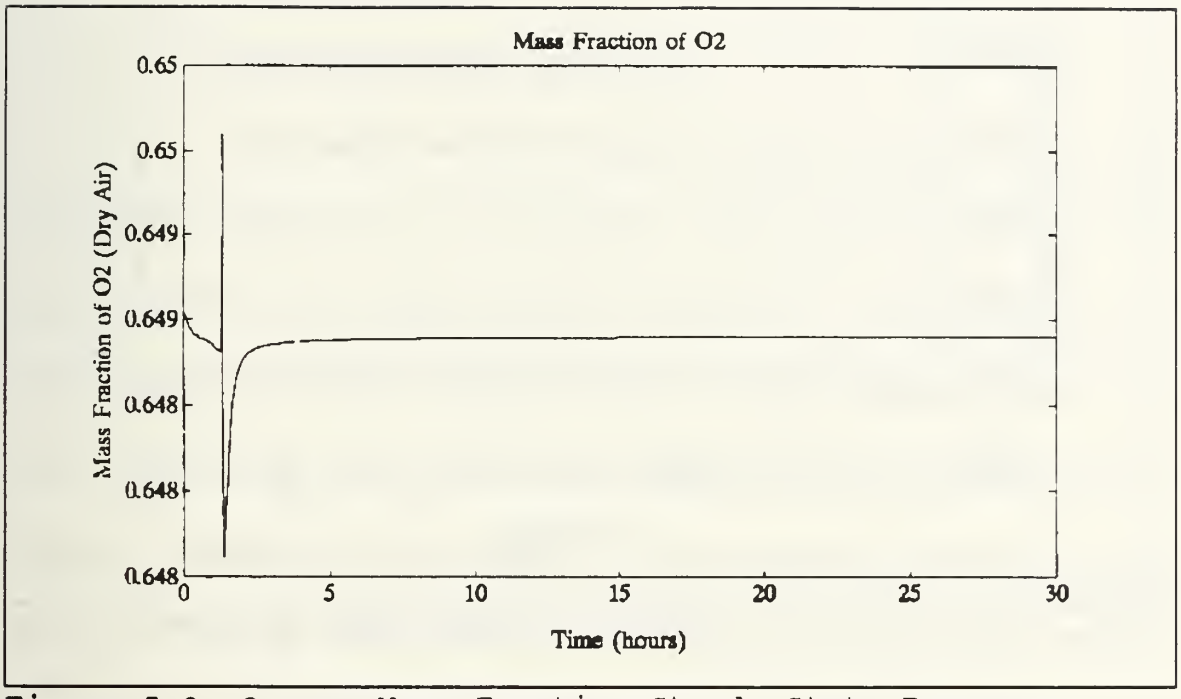


Figure 5.2 Oxygen Mass Fraction Steady State Response

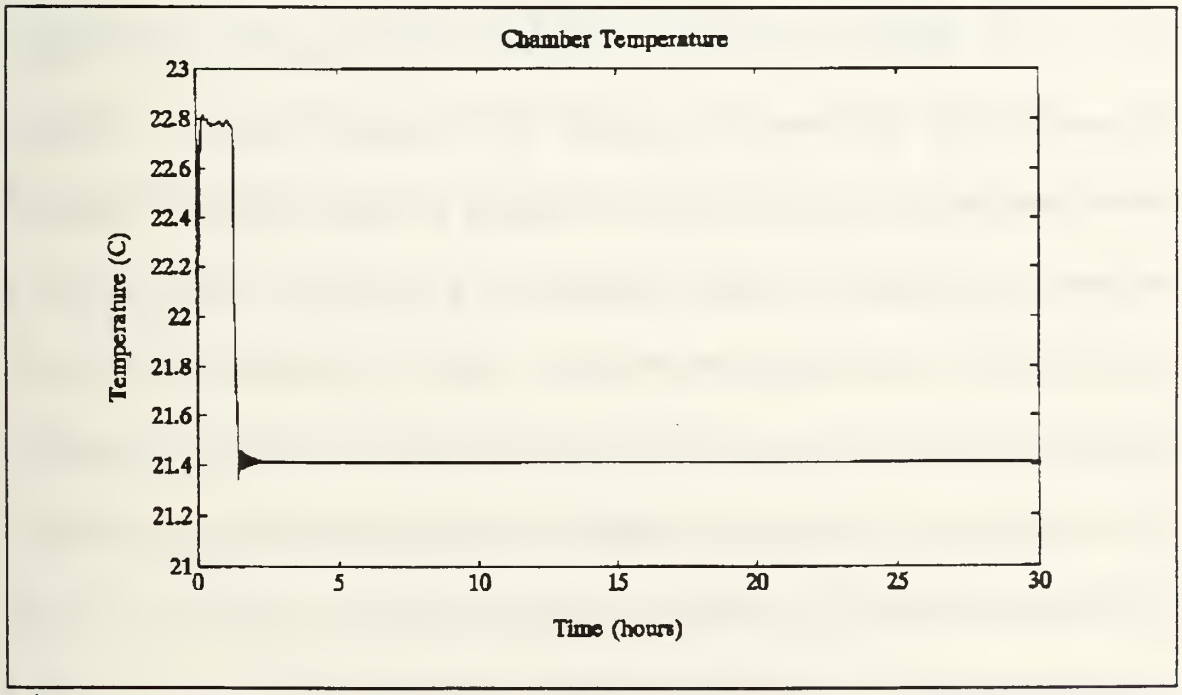


Figure 5.3 Temperature Steady State Response

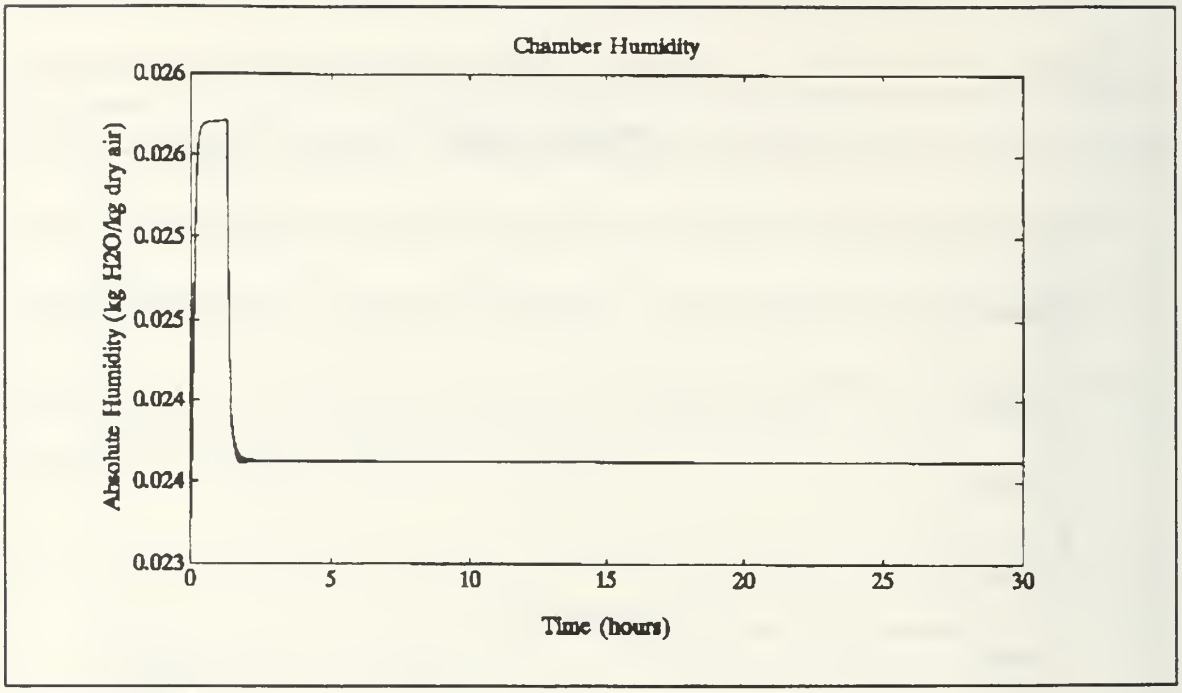


Figure 5.4 Absolute Humidity Steady State Response

2. Step Response

In addition to stability, sensitivity and accuracy, we are always concerned with the transient response of a feedback system. Transient characteristics are normally defined on the basis of a step response. Therefore, the next performance evaluation undertaken is a simulation of the system in a transition from a state defined at 34.464 kPa to one defined at 51.69 kPa. Transition control in a hypobaric environment is critical. The key to this transition lies in the ability of the system to maintain the mass fraction (or partial pressure) of oxygen in the proper proportion to the total mass (or total pressure). The physiological effects of an improper transition to lower pressures are great and

must be addressed in any control algorithm. For the transient response, both the direct and indirect adaptive controllers will be investigated.

a. Indirect Controller Performance

The adaptive based linear quadratic regulator controller performance is provided in Figures 5.5, 5.6, 5.7 and 5.8. The pressure and oxygen mass fraction transitions are adequate (Figures 5.5 and 5.6). The transitions are both essentially linear and achieve the new steady state at approximately the same time with minimum over shoot. This is an indication that the proper proportion of oxygen is maintained during the transition. Since an upward transition in pressure does not involve any physiological considerations (i.e., the bends) and there are no current experimental time constraints imposed, the settling time is not considered a critical performance measure. The main deterrent to more rapid transition is the physical constraints imposed by the makeup flow rates. The temperature transient response is presented in Figure 5.7. A comparatively large jump is observed at the onset of the set point change. This may be due in part to the coupling of the state variables. Pressure and oxygen mass fraction carry a much greater weight in the control law development, ie. the Q matrix in the LQR algorithm is weighted significantly towards these variables. As a result. the controller initiated an immediate increase in pressure. In conformance with the ideal gas law for a fixed volume, an increase in pressure is accompanied by an

increase in temperature. Since the temperature set point did not change in the transition, its initial response is only a result of the changes initiated for humidity and pressure. The humidity transition to the new set point is adequate with a minor spike encountered at the outset, largely due to an initial jump in temperature. Numerous follow on simulations were conducted to reduce this temperature anomaly by adjusting the weighting of the Q and R matrices with little success. The steady state errors at the new set point exhibit minimal variance from the errors observed at the initial set point.

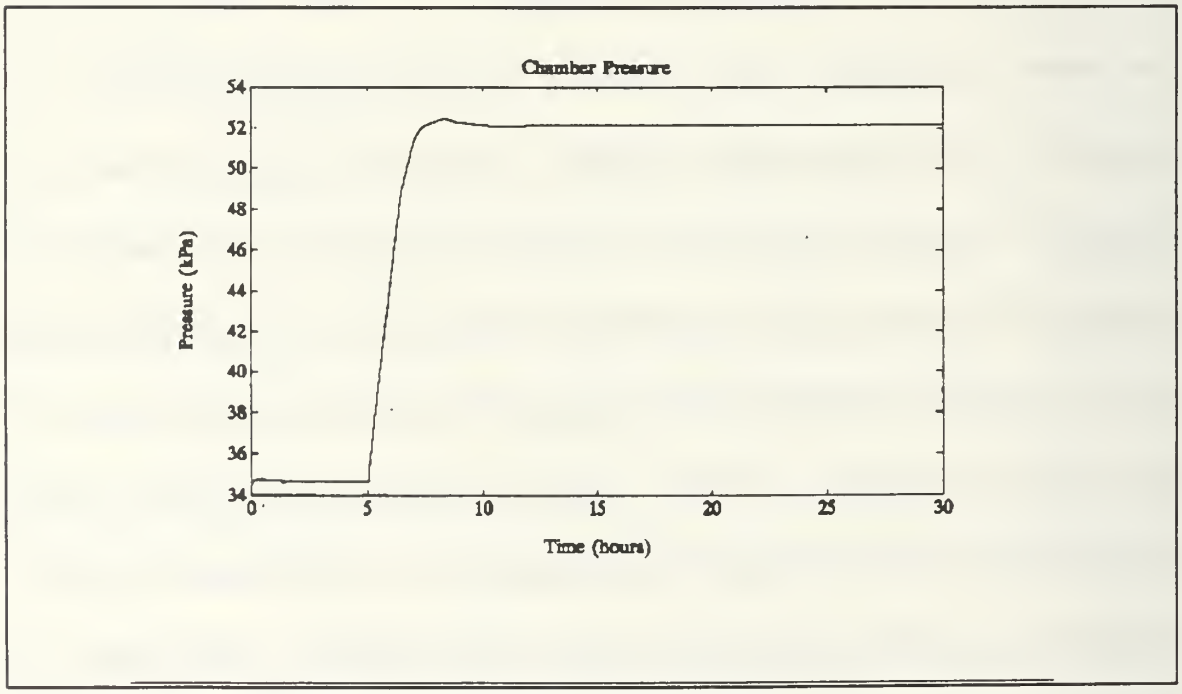


Figure 5.5 Pressure Transient Response

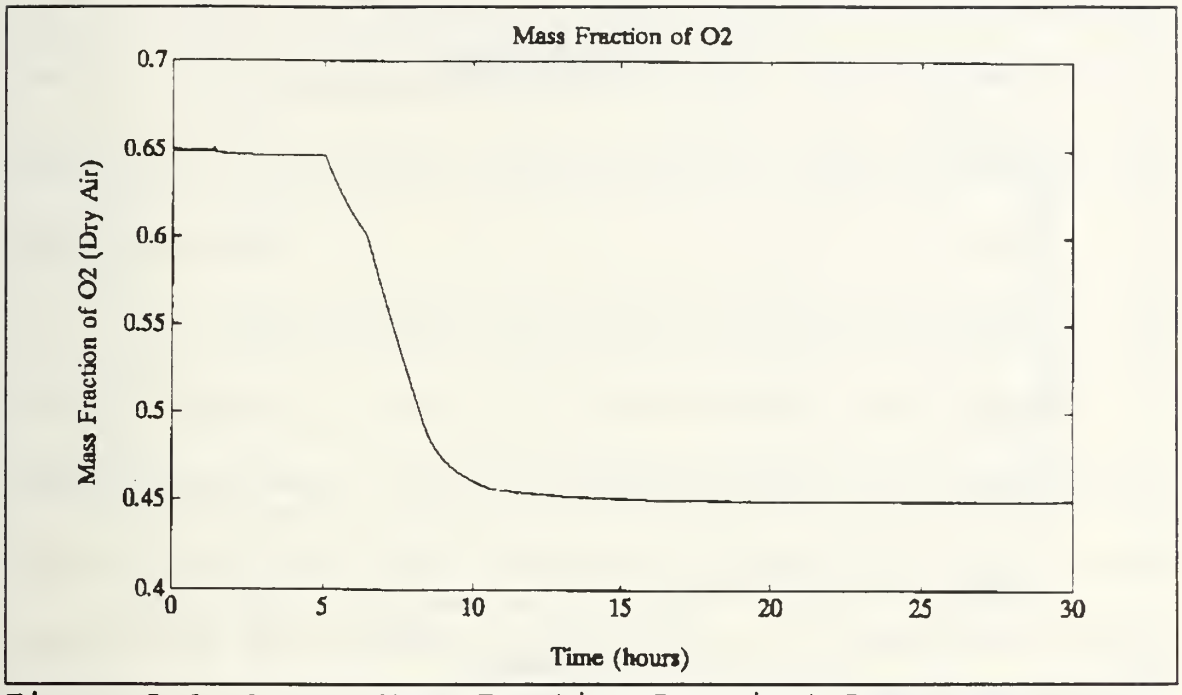


Figure 5.6 Oxygen Mass Fraction Transient Response

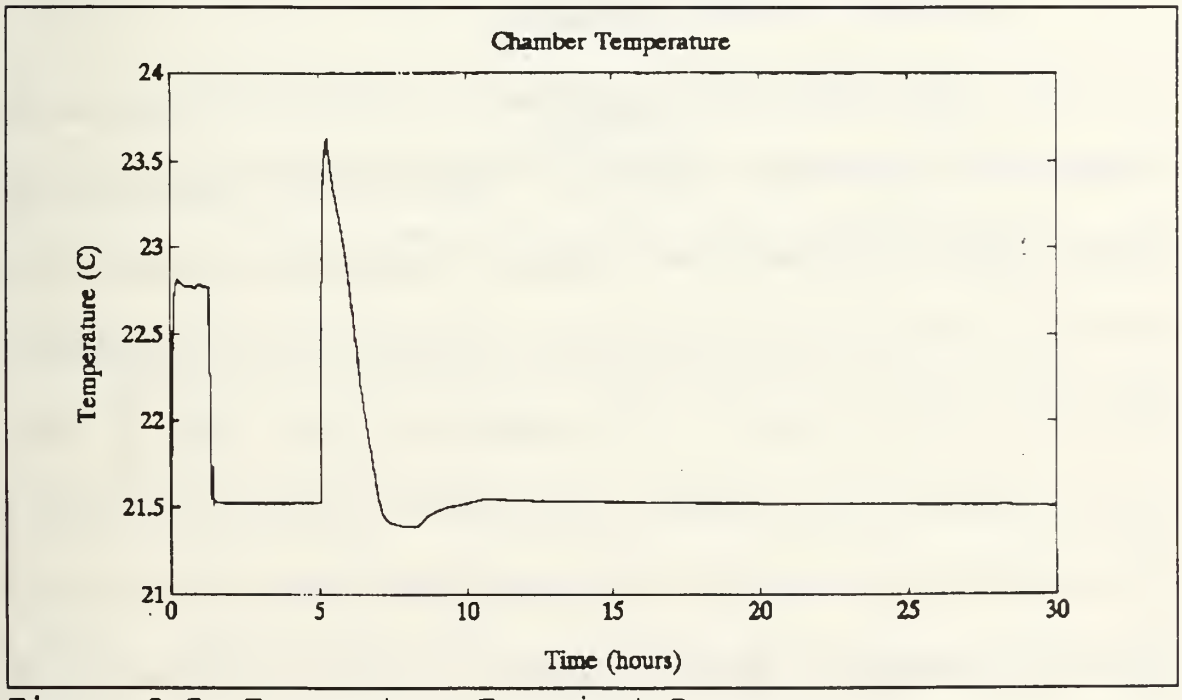


Figure 5.7 Temperature Transient Response

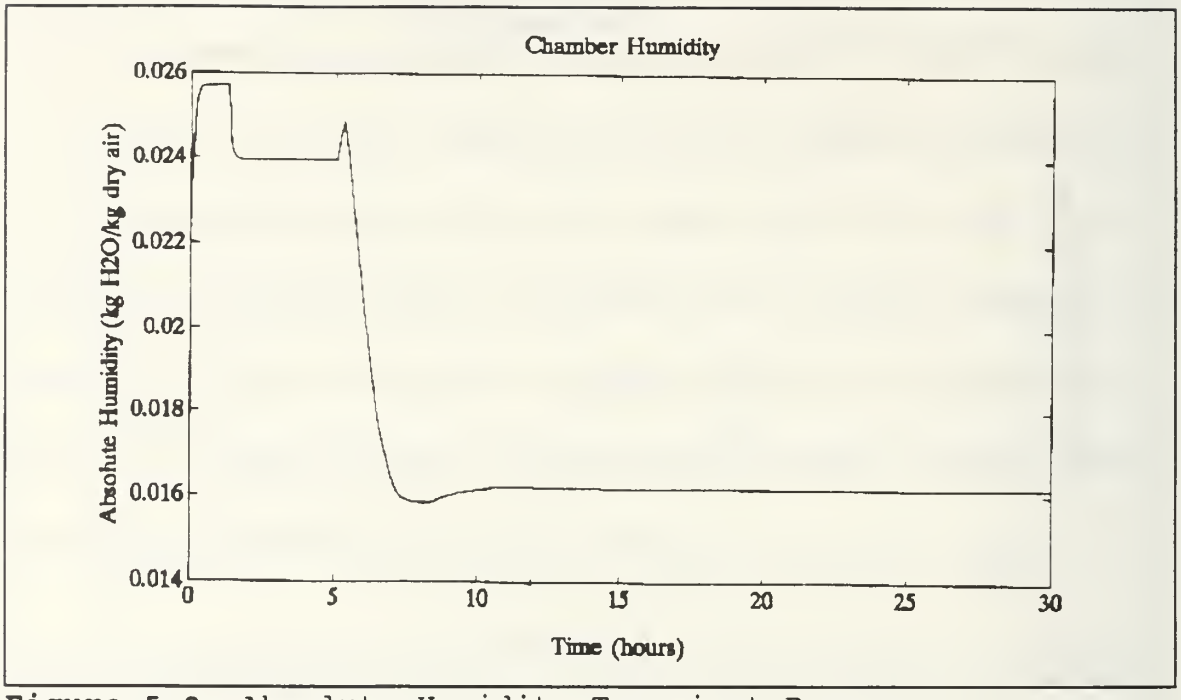


Figure 5.8 Absolute Humidity Transient Response

b. Direct Controller Performance

The direct control algorithms are written in Fortran and the computer code is presented in Appendix A. These have been integrated into the Matlab model for simulation and analyses. The transient response of the predictive, multivariable control algorithm to a change in set point is presented in Figures 5.9, 5.10, 5.11, and 5.12. The responses are very similar to those obtained from the LQR controller. The transitions in pressure and oxygen is almost identical in both shape and settling time (Figures 5.9 and 5.10). There was slightly less overshoot in pressure transient. In terms of temperature, the pertubations about the set point are comparatively smaller (Figure 5.11). A spike

of only about 1 °C was observed, compared to over 2 °C for the indirect controller. The humidity response in the direct algorithm represents an improvement over the indirect response, largely due to the minimization of the temperature variance.

The steady state errors for the new set point are near zero. This can be accounted for in part to the computationally intensive simplex and Diophantine algorithms. The 30 hour simulation required over 24 hours of actual computer time on a VAX system, compared to only about 15 minutes for the LQR based controller on a IBM 80486 personal computer. The computational intensity of the direct controller severly limits it viability. Computer code streamlining and optimization would be required for implementation.

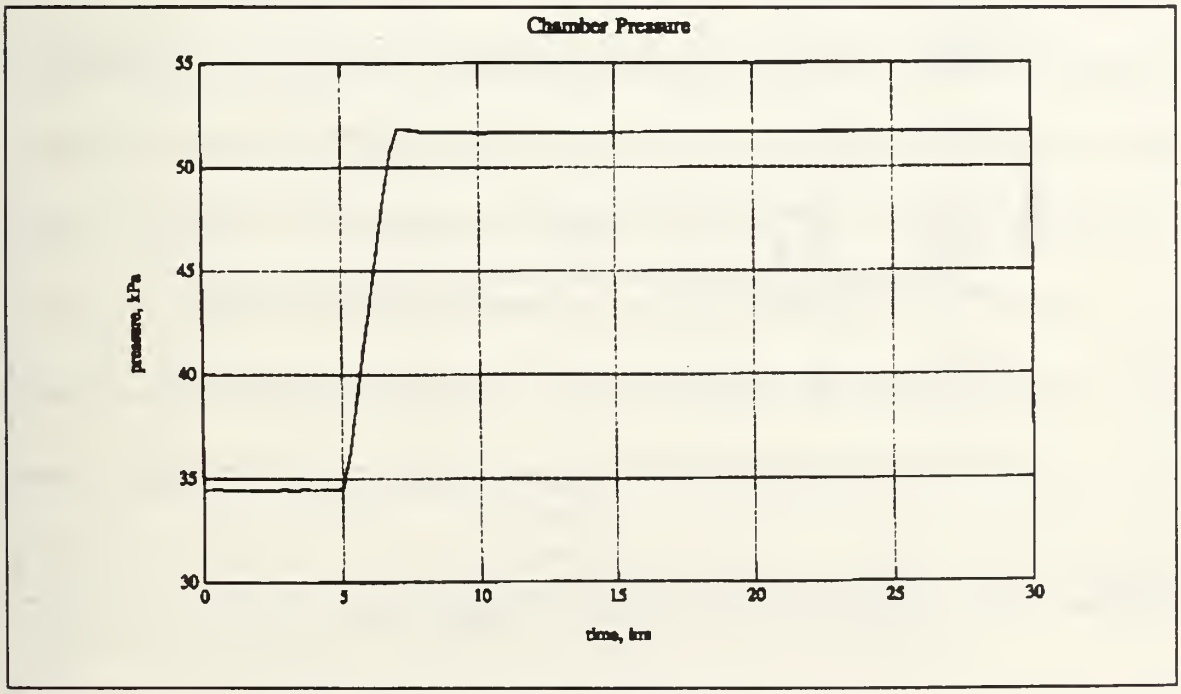


Figure 5.9 Pressure Transient Response

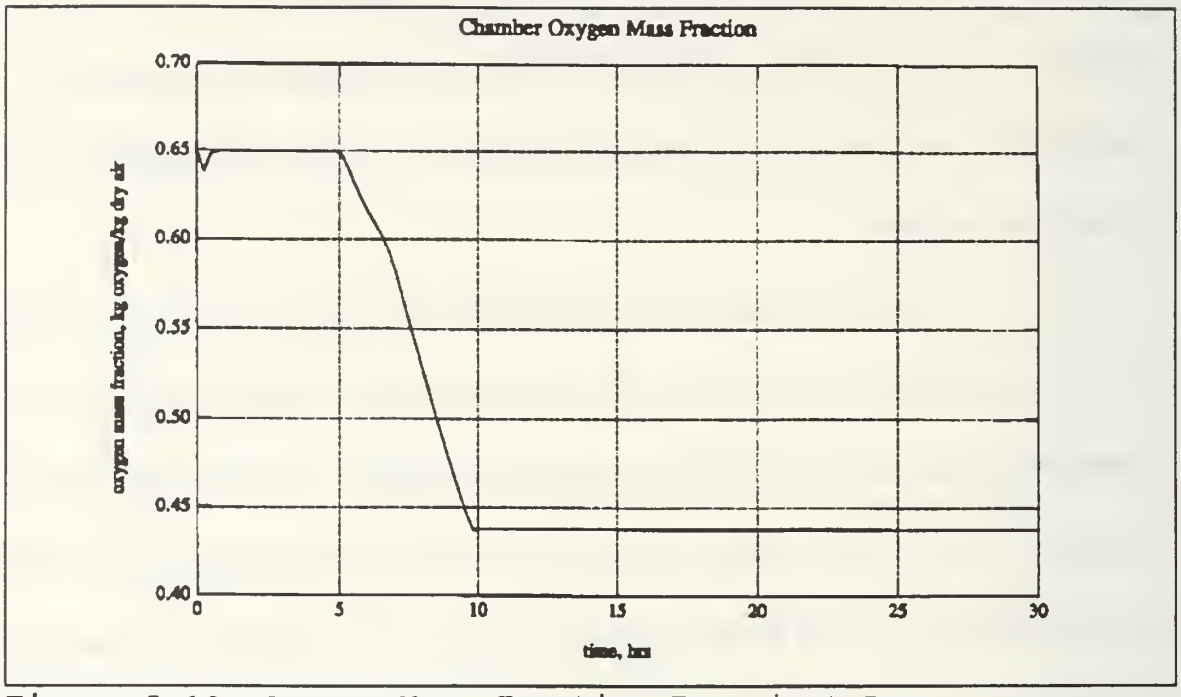


Figure 5.10 Oxygen Mass Fraction Transient Response

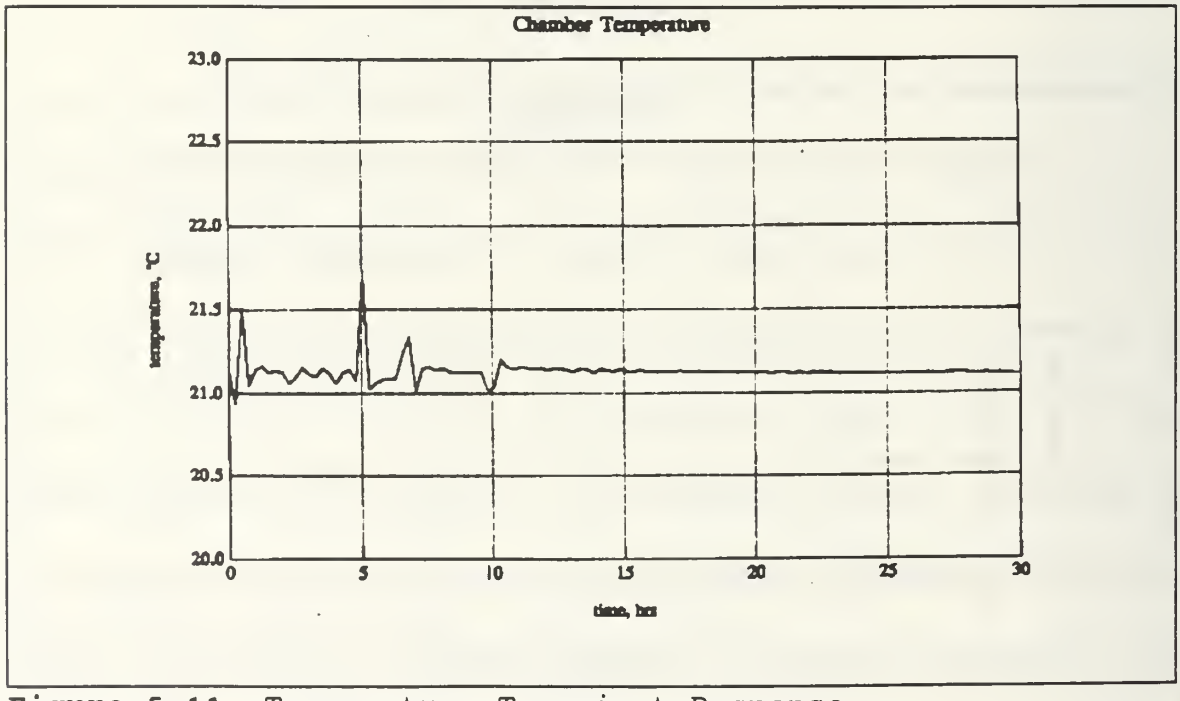


Figure 5.11 Temperature Transient Response

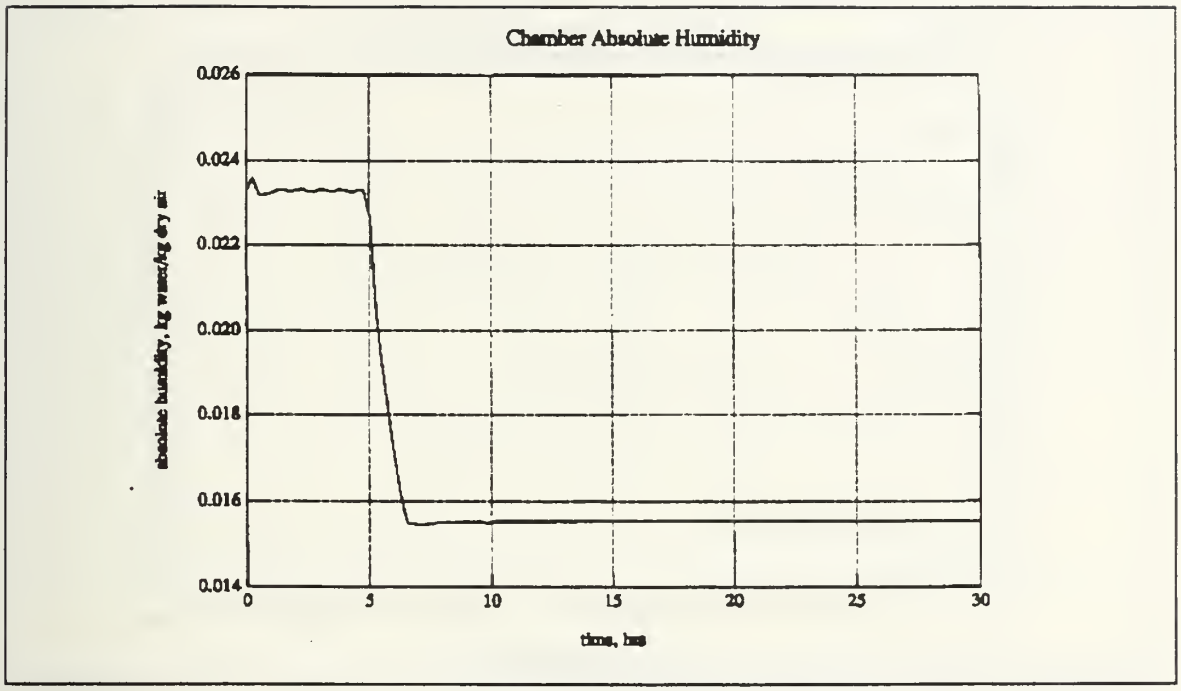


Figure 5.12 Absolute Humidity Transient Response

3. Load Disturbance

An effective manner in which to test the adequacy of the control algorithm is to subject it to demanding loading conditions. Since the range of potential loads in the CERC can vary significantly depending upon the prevalent human acitivity, it is necessary to subject the controller to a large variation in loading. Hence, the response of the LQR based controller to a series of step disturbance increases was analyzed. For this analysis, the latent and sensible heat loads in additon to the oxygen consumption have been increased incrementally over a 50 hour simulation at the same operating set points. A graphical representation of the variable loading is available in Figures 5.13, 5.14 and 5.15.

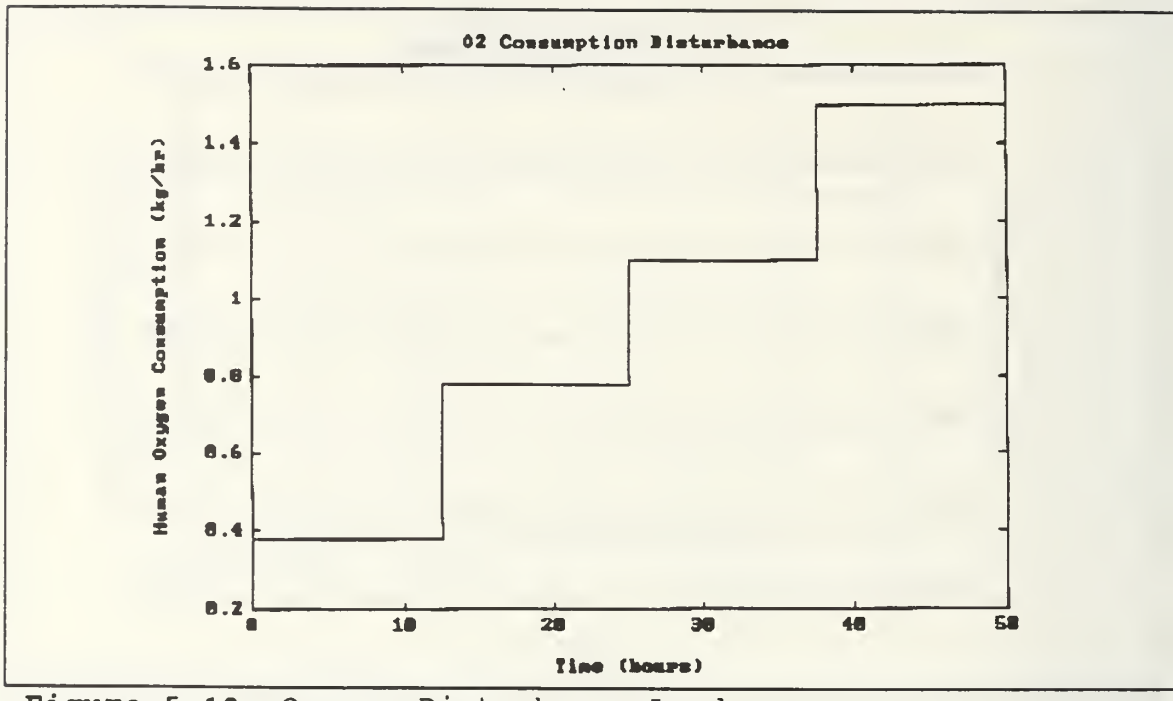


Figure 5.13 Oxygen Disturbance Load

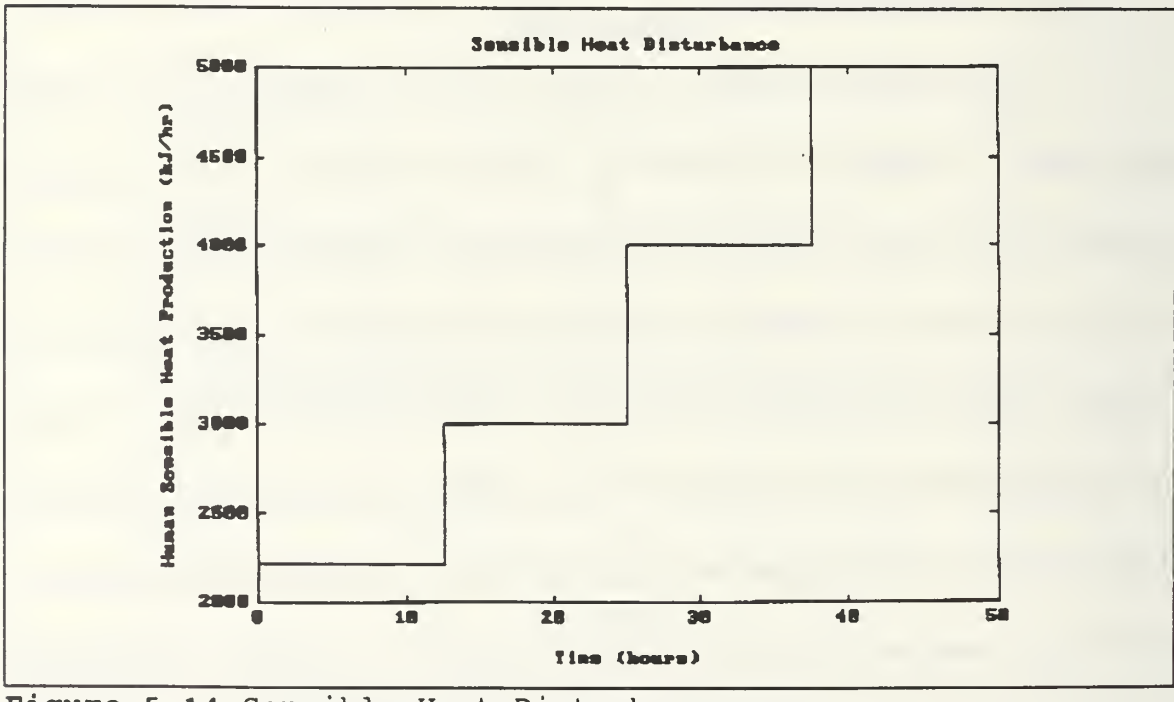


Figure 5.14 Sensible Heat Disturbance

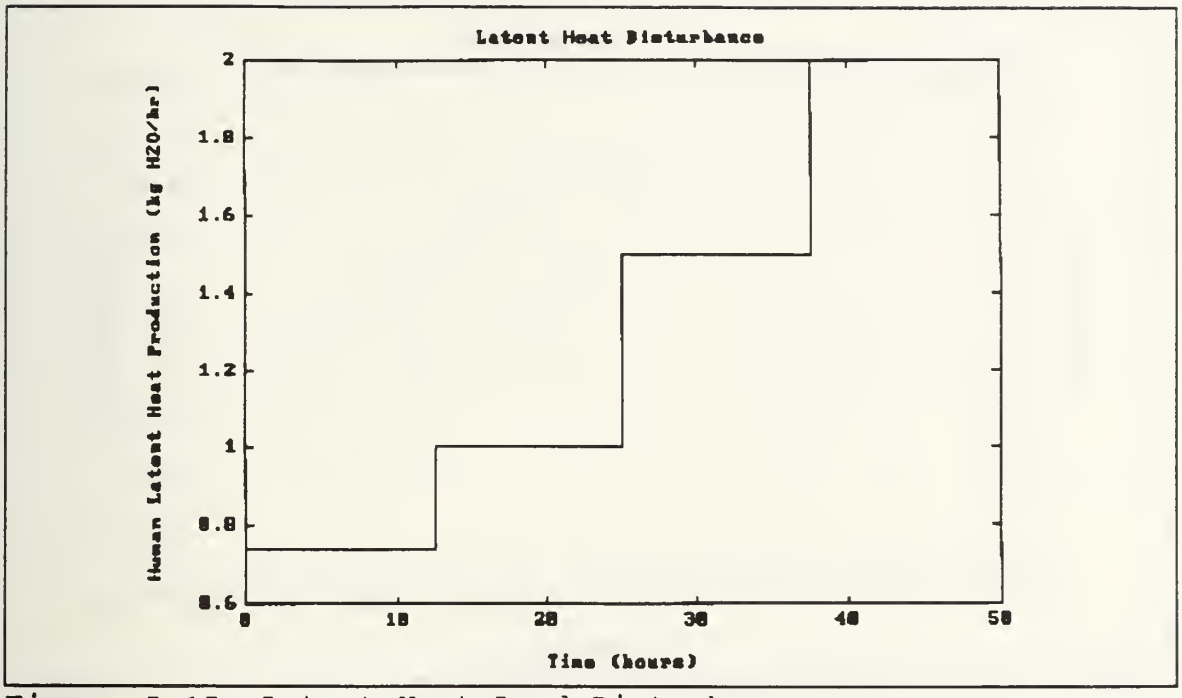


Figure 5.15 Latent Heat Load Disturbance

In order to accurately appraise the performance of the closed loop controller, it is necessary to analyze the open loop system response to the incremental disturbance. This response is presented in Figures 5.16, 5.17, 5.18 and 5.19. The nonlinearity of the dynamic system can be readily observed in the first order response.

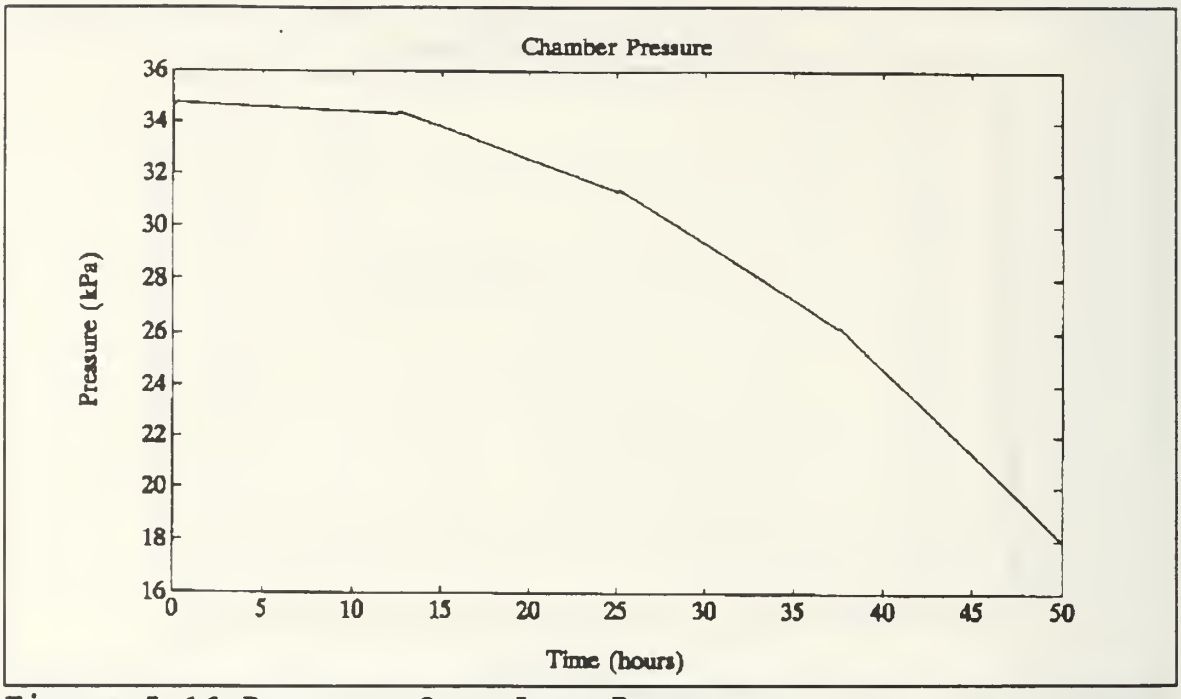


Figure 5.16 Pressure Open Loop Response

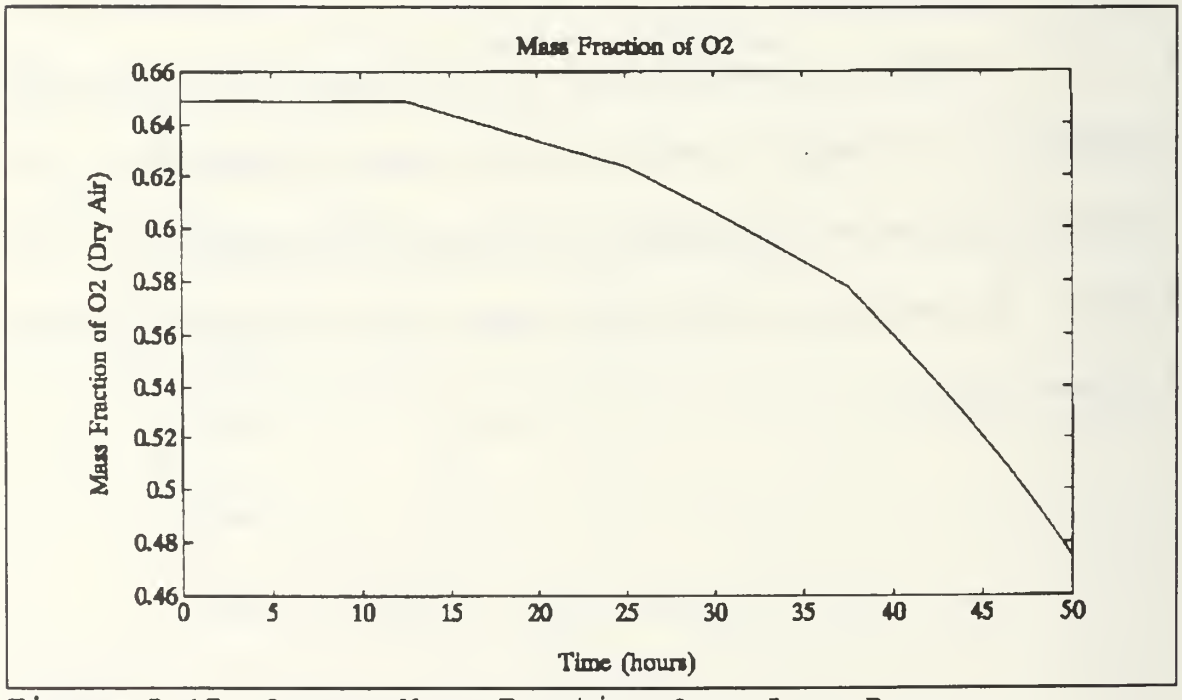


Figure 5.17 Oxygen Mass Fraction Open Loop Response

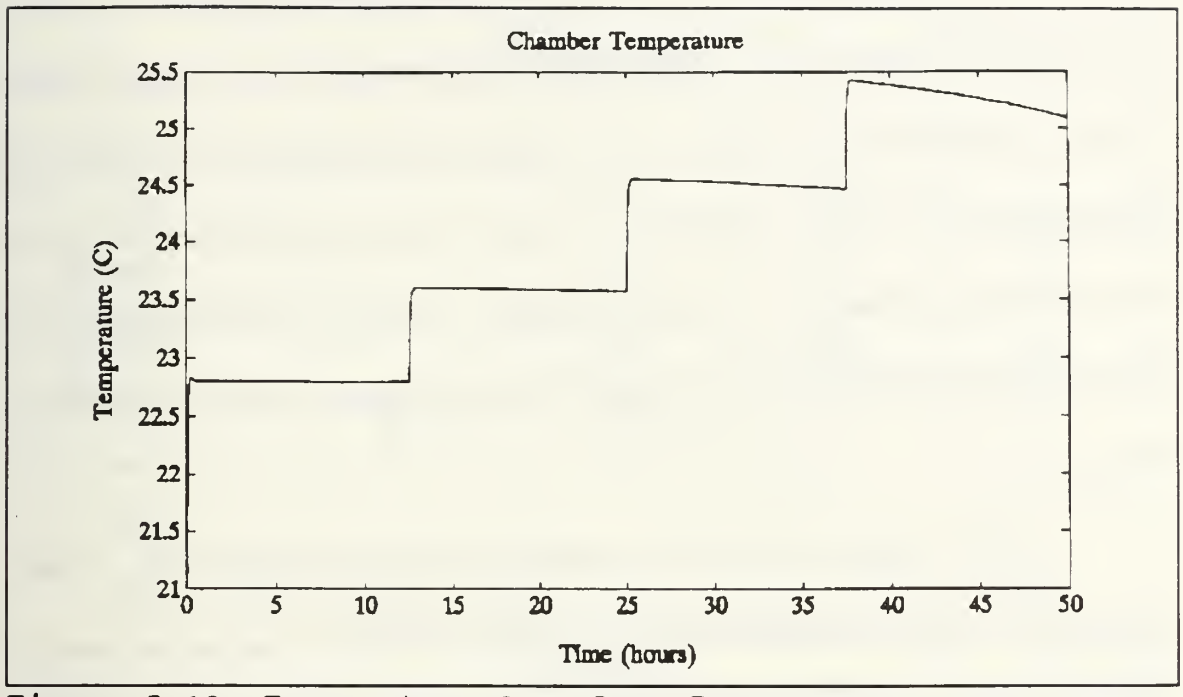


Figure 5.18 Temperature Open Loop Response

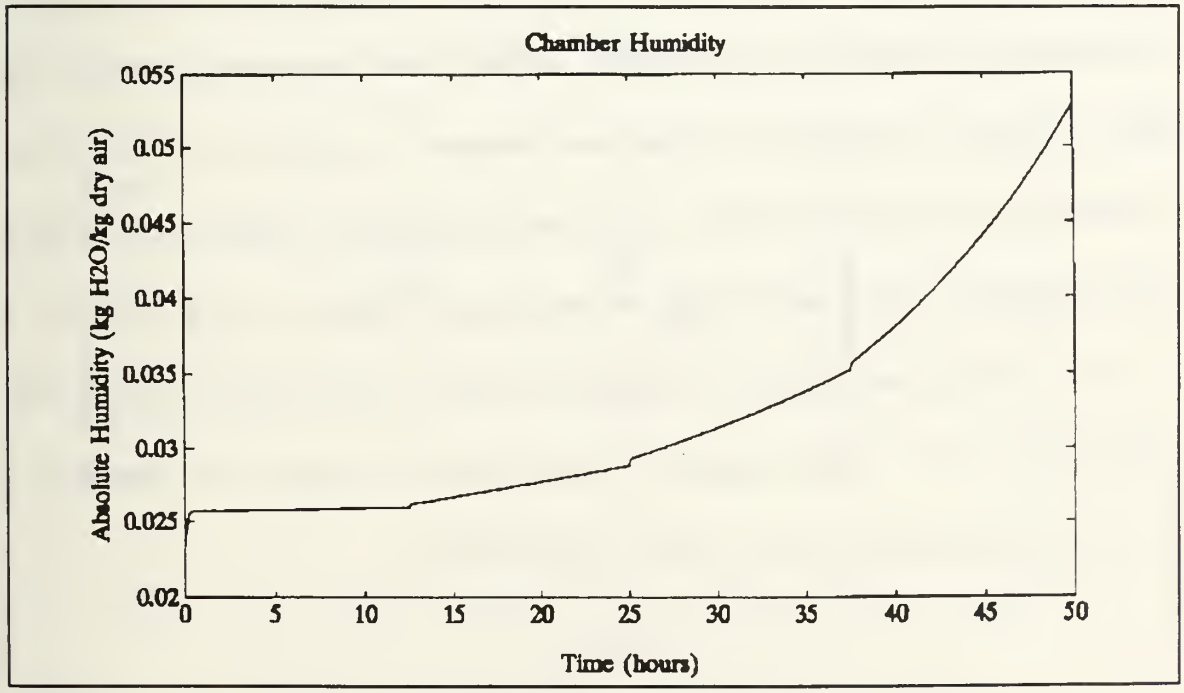


Figure 5.19 Absolute Humidity Open Loop Response

The load disturbance on the closed loop system is depicted in Figures 5.20, 5.21, 5.23 and 5.24. Looking specifically at the pressure response (Figure 5.20), it is observed that only a series of spikes corresponding to the onset of the disturbances deters from a relatively flat curve. The maximum pertubation from set point only represents a 1.4 percent variance. In light of the magnitude of the load disturbance from assumed steady state, the response may be classified as adequate.

The oxygen mass fraction response is excellent (Figure 5.21). Despite tripling the oxygen consumption and latent heat production, the largest observed deviation from set point is just 0.3 percent. Temperature variation, depicted in Figure 5.22, is comparatively large. The relative weighting of the temperature, as discussed previously, is one probable cause of the larger deviation from set point. Efforts to improve the temperature response by adjusting the weight had a tendency to cause the controller to cycle and go unstable. Despite over doubling the overall latent and sensible loads, the maximum variation was still limited to 2 °C. The tracking and control of humidity, because of the strong coupling with temperature, had a similar response. The maximum overshoot was observed to be 0.002 kg water/kg dry air, or about eight percent.

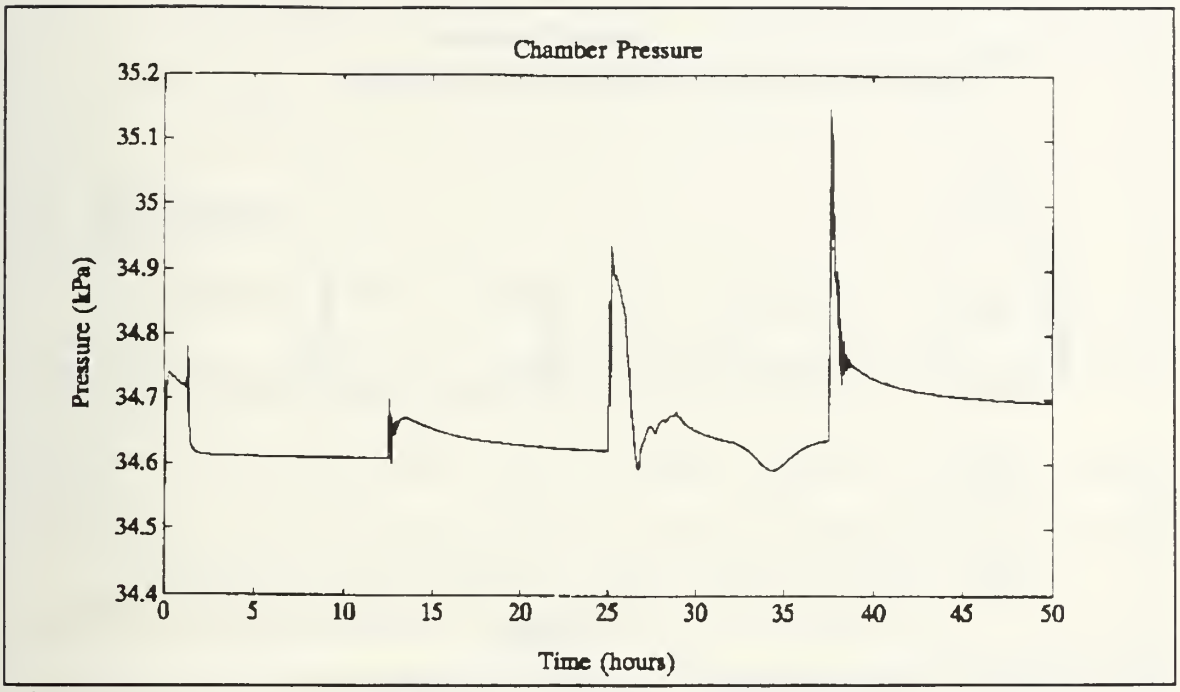


Figure 5.20 Pressure Closed Loop Disturbance Response

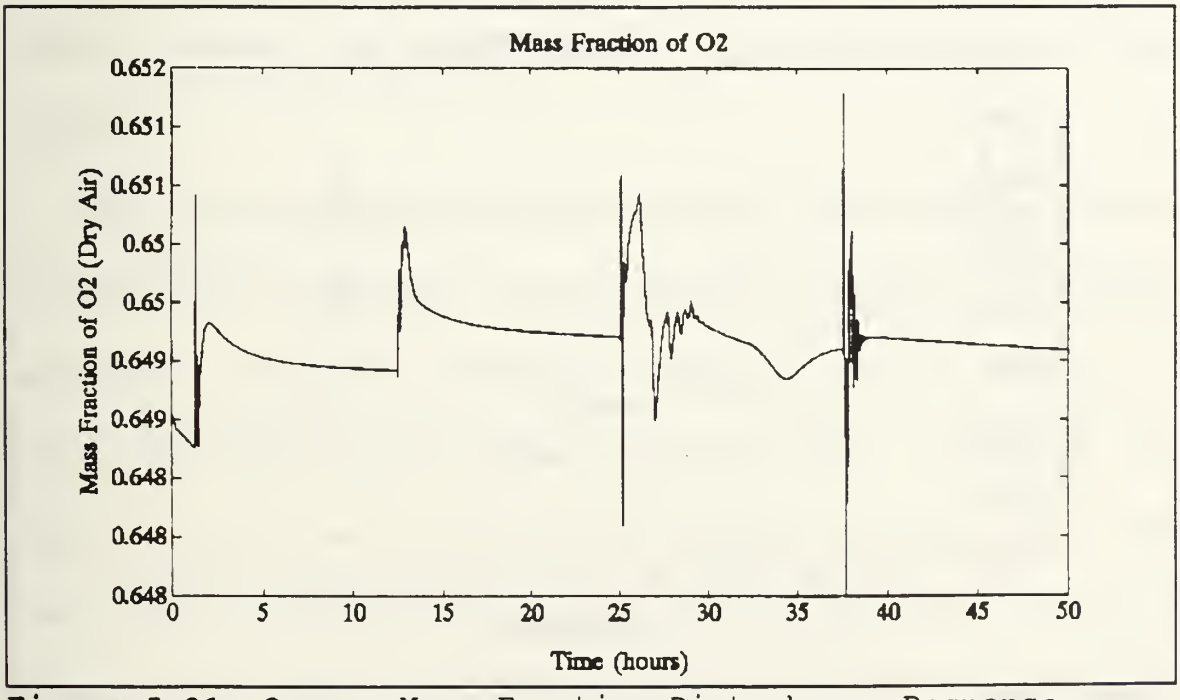


Figure 5.21 Oxygen Mass Fraction Disturbance Response

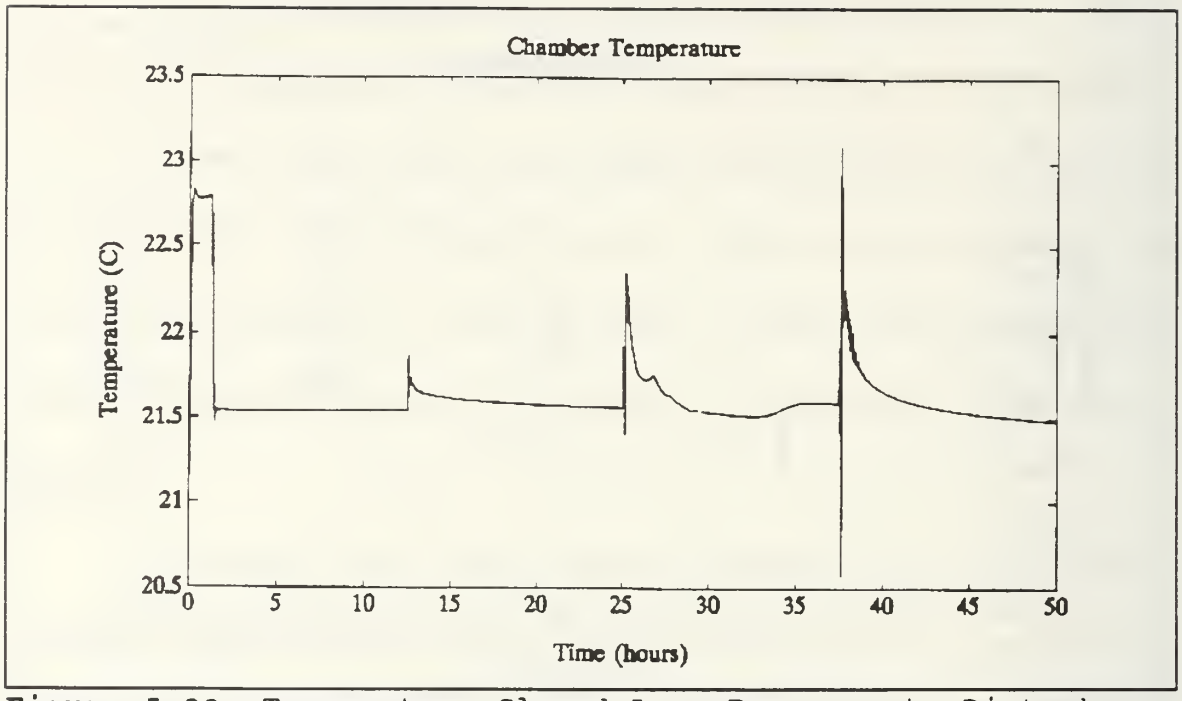


Figure 5.22 Temperature Closed Loop Response to Disturbance

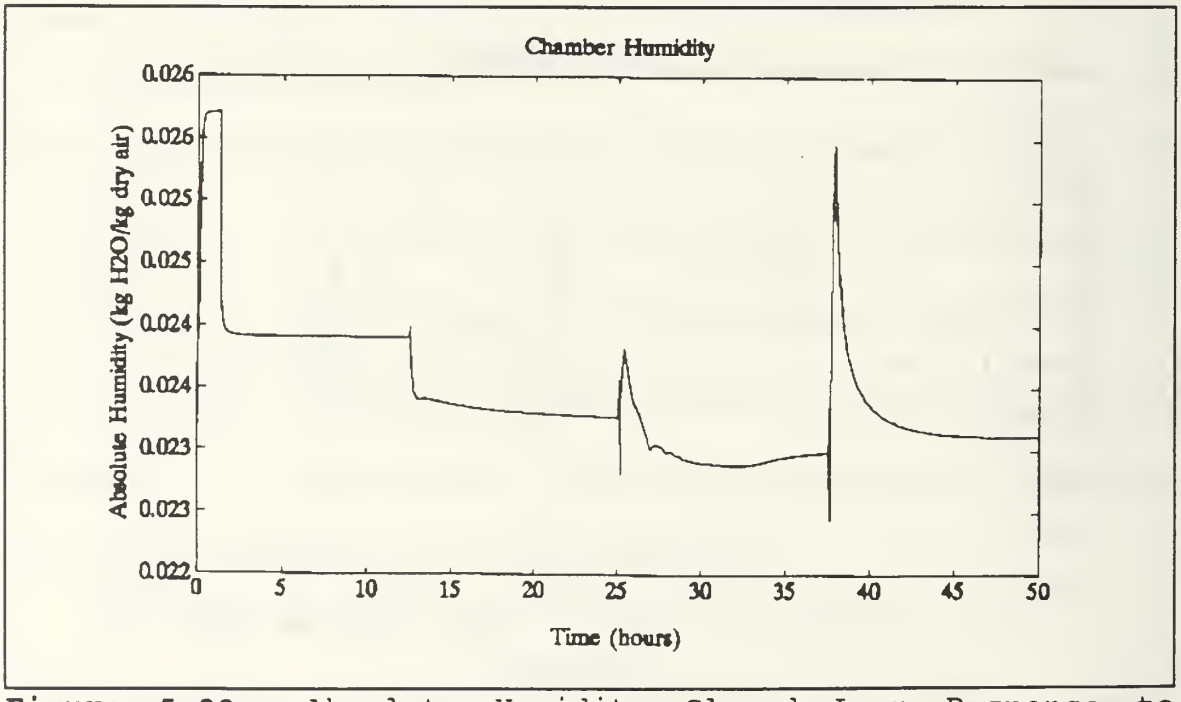


Figure 5.23 Absolute Humidity Closed Loop Response to Disturbance

VI. CONCLUSIONS/RECOMMENDATIONS

A. SUMMARY

A preliminary control algorithm and mathematical model have been developed for the atmospheric system within the Controlled Environment Research Chamber located at NASA Ames Research Center. The basic system configuration and groundwork parametric studies have been reviewed for inclusion in the basic modeling equations. Adaptive control techniques have been developed and tested for potential utilization in system identification and control of this nonlinear, multivariable process. Included within these techniques are algorithms involving recursive least squares estimator for the approximation of system dynamics.

The indirect adaptive control scheme, utilizing a linear quadratic regulator for control law formulation, provided satisfactory control of the chamber atmosphere. The controller effectively maintained system state variables in a steady state simulation with minimal steady state error. Additionally, the algorithm tracked a set point change with essentially no overshoot and a finite steady state error. The introduction of significant metabolic step disturbances in system operation caused only small deviations in the atmospheric set points.

The direct control algorithm, adapted in cooperation with Dr. Cory Finn, exhibited a greater degree of accuracy in set point maintenance. The transient response to a set point change was similar in form to the indirect algorithm. However, it was computationally intensive and in its present configuration is not a viable alternative.

Presuming the model developed adequately portrays the chamber dynamics, adaptive control techniques appear to be an excellent alternative for the simultaneous control of temperature, pressure, composition and humidity.

B. RECOMMENDATIONS

Significant additional work is necessary prior to implementation of this adaptive control algorithm. The following is a listing of areas which will require attention:

- The mathematical model developed requires the inclusion of the dynamics of the control system actuators, measurement and metering devices and associated time delays. In addition, mass and energy diffusion dynamics should be analyzed for potential inclusion into the model.

- Strawman experiments involving chamber utilization must be refined to accurately define control system requirements. These experiments would provide transient response characteristics required as well as a better representation of overall operating envelopes.

- Increased work on hardware selection and parametric performance studies. In particular, a suitable condensing heat exchanger must be selected and performance criteria implemented into the chamber dynamic model. Additionally, performance data needs to be refined for the gas supply system.

- Additional simulations need to be conducted on the improved model to optimize performance of the algorithm. Eventual implementation into the chamber requires exhaustive utilization of preliminary testing to maximize the performance.

APPENDIX A. COMPUTER PROGRAMS

1. This first section details the model and the indirect adaptive control routine that utilizes an LQR control gain calculation.

```
%*************************************************************************
%                                                    Program cerc.m
%*************************************************************************

!del bhm1.met
!del bhm2.met
!del bhm3.met
!del bhm4.met
clg;
clear;

%*********************************************************************
%   This computer code represents a dynamic model and  indirect adaptive control
%   routineof the temperature, composition, pressure and humidity of the habitat
%   enclosedwithin the  Closed Environment Research Chamber that is currently
%   undergoingrenovationat NASA Ames Research Center located at Moffet Field,
%   CA
%*********************************************************************

%   The variables utilized for state definition and control are as follows:

% x1 = Chamber pressure (kPa)
% x2 = Mass Fraction of O2 in dry air
% x3 = Chamber Temperature (degrees C)
% x4 = Chamber absolute humidity (kg H2O/kg dry air)
% u1 = Vacuum flow rate out of chamber (kg/hr)
% u2 = Mass flow rate of oxygen into chamber (kg/hr)
% u3 = Mass flow rate of nitrogen into chamber (kg/hr)
% u4 = Coolant stream entrance temperature into CHX (degrees C)
% u5 = Air flow rate from chamber into CHX (kg/hr)

%    The following are variables that are utilized in system defintion:

% U = Heat exhanger performance parameter (KJ/m^2-C)
% A = Heat exchanger area  (m^2)
% mc = Coolant flow rate across CHX (kg/hr)
% R = Universal gas constant  8.314 kJ/kgmole K
% Vc = Chamber volume (m^3)
% mleak = Leakage of ambient air into the chamber (kg/hr)
% x1 = Mass fraction of O2 in ambient (outside) air
% O2 = Human consumption of O2 (four persons) (kg/hr)
```

```
%  H2O= Human production of H20 (four persons) (kg/hr)
%  qh= Human sensible heat load (four person)(kJ/hr)
%  qm= Machinery/miscellaneous sensible heat loads (kJ/hr)
%  a1,a2,a3,b1,b2,b3= Coefficients of heat capacities for
%           chamber constituents (O2,N2,H20)
%  vap= heat of vaporization of H2O from human in chamber-2287 kJ/hr
%  hvap1= Heat of vaporization of H2O in CHX exit stream
%  Q= Matrix constant
%  x3= Chamber temperature in degrees K
%  x4= CHX air exit temperature in degrees K

%    The simulation of the multivariable, non linear equations will be
%    set up as follows:

%    xdot= F(x)*xdot + G(x,u) + H(x,w);

%   Solving for xdot :

%     xdot= (I-F(x))-1*G(x,u) + (I - F(x))-1* H(x,w)

%   where x is a vector of state variables, u is a vector of the control inputs
%   and w is a vector of the metabolic loads.

%*********************************************************************

%    Setting the initial conditions of the states and control variables for a simulation
%    involving a state defined at 34.464 kPa.  In addition, the set point at this state
%    is given as well as the maximum control inputs.

 x=[34.464; 0.64903;21.1;0.02327391];
 xset=[34.464;0.64903;21.1;0.0232739];
 u=[1.0;1.0;0.3658;3;980];
uset=[1;1.00;0.3658;1.5;980];
umax=[10;10;10;4;2000];

%*********************************************************************

%    Setting the other variables

%    Estimated volume of the chamber  (m3)
Vc=157;
%    Heat capacity coefficients (1-O2, 2-N2, 3-H2O)
a1=0.9324;
a2=0.90468;
a3=1.831;
b1=0.000184;
b2=0.000204;
b3=0.000744;

%    Machinery/miscellaneous sensible heat load inside chamber (kJ/hr)
qm=8586.33;
%    Oxygen consumption of humans (4 persons-3@normal activity,
%    1@exercise)  (kg/hr)
O2=0.3956;
%    Estimated Latent heat production - humans (water) (kg/hr)
H2O= 0.7398;
%    Estimated Sensible heat production - humans (kJ/hr)
qh=2213;
%    Standard heat of vaporization of water inside chamber (@ 21.1oC)
vap=2287;
```

```
%    Estimated air leakage into chamber  (kg/hr)
mleak=0.0025;
%    Oxygen mass fraction of air leaking into the chamber
x1=0.2344;
%    Universal gas constant  (kJ/kgmole-K)
R=8.314;

 %*****************************************************************
 %   Initializing the vectors for utilization in the recursive least squares algorithm

alpha=0.0001;
q1=alpha*eye(9);
th1=zeros(9,1);
q2=alpha*eye(9);
th2=zeros(9,1);
q3=alpha*eye(9);
th3=zeros(9,1);
q4=alpha*eye(9);
th4=zeros(9,1);

%*****************************************************************
%    Setting the overall simulation length, integration time step, and sampling time

%    Overall number of sampling time steps
n=2000;
%    Sampling instance (i.e. p integration time steps per sample)
p=6;
%    Integration time step (hrs)

%    T is the sampling duration required in the integration of the equations
%    of motion that define the dynamics of the processes involved in
%    regulating temperature,  humidity, pressure and composition.
%    Simulation for n integration steps
T=0.004167;

%    Setting up the time vector
time=[1:n*p+1];
time=time*T;
%    Running the simulation for n samples
for k=1:n;



%    The simulation will run for p integration time steps prior to sampling
%    and evaluating the data for parameter estimation and control input.
p=6;
r=p-1;
for i=p*k-r:p*k

x1=x(1,i);
x2=x(2,i);
x3=x(3,i);
x4=x(4,i);
x5=x(3,i)+273;
x6=u(4,i)+281.33;
x7=u(4,i)+8.33;
u1=u(1,i);
u2=u(2,i);
u3=u(3,i);
u4=u(4,i);
```

```
u5=u(5,i);

%*********************************************************************
%    Change in the set points at t=5 hours
%    if i>1200
%    xset=[51.69;0.444653;21.1;0.015537];
%    end;
%*********************************************************************


%*********************************************************************
%    Incremental increases in metabolic loads

%    if i>3000
%    H2O=1;
%    qh=3000;
%    O2=0.8;
%    end;
%    if i>6000
%    H2O=1.5;
%    qh=4000;
%    O2=1.1;
%    end;
%    if i>9000
%    H2O=1.7;
%    qh=5000;
%    O2=1.5;
%    end;

%*********************************************************************

%    Calculate the partial pressure of H2O in the exit stream of the
%    CHX

Pstar=10^(28.5905-8.2/2.306*log(x6) + 0.002484*x6 - 3142/x6);

    if x3<x7
        Pstar=0.0;
    end;

%    Calculate the heat of vaporization of the moisture in the exit
%    stream of the CHX

Hv=2213;
hvap1=(13.11 - log(Pstar*0.9869))*R/18.02*x6;

%    Calculate the humidity of the CHX exit stream

we=18.02/(4*x2+28+18.02*x4)*(100*Pstar/(x1-100*Pstar));


%    Calculate the enthalpies of the entrance and exit gases of the CHX

CpT1=x2*(a1*x3 + b1*x3^2) + (1-x2)*(a2*x3 + b2*x3^2) ...;
+ x4*( a3*x3 + b3*x3^2);
CpT2=x2*(a1*x7 + b1*x7^2) + (1-x2)*(a2*x7 + b2*x7^2) ...;
 + we*(a3*x7 + b3*x7^2);

%    Calculate the change in sensible heat removed across the CHX
```

```
delh=CpT2 - CpT1;

%     Calculate the change in absolute humidity across the CHX

delw=we -x4;

    if delw >0
        delw=0.0;
    end;

%     Calculate the latent heat removed across the CHx

dell=delw*hvap1;
delh20=u5/(1+x4)*delw;

%     Calculate the total heat removed across the heat exchanger

delchx=u5/(1+x4)*(delh+dell);

%     Calculate the function Q

Q=(x2*(a1 - a1*x3/x5 + 2*b1*x3 - b1*x3^2/x5) + (1-x2)*(a2 - ...;
a2*x3/x5 + 2*b2*x3 - b2*x3^2/x5) + x4*(- Hv/x5 + a3 - a3*x3/x5 + ...;
 2*b3*x3 - b3*x3^2/x5));

%*********************************************************************
%     The following code calculates the change in the state variables given previous
%     states and current input

%     The following is the mathematical representations of the matrices
%     F, G, and H.



F=[0,0,x1/x5,0; -x2/x1,0,x2/x5,x2/(1+x4);-(x2*(a1* ...;
x3 + b1*x3^2) + (1-x2)*(a2*x3 + b2*x3^2) + x4*(Hv + a3*x3 + ...;
b3*x3^2))/(x1*Q), -(a1*x3 +b1*x3^2 - a2*x3 - b2*x3^2)/ ...;
Q, 0,-((-1/(1 + x4))*(x2*(a1*x3 + b1*x3^2)+ (1 - x2)* ...;
(a2*x3 + b2*x3^2)) + Hv + a3*x3 + b3*x3^2 - (x4/(1+ x4))*(Hv + a3*...;
x3 + b3*x3^2))/Q; -1/(x1*(1/x4 - 1/(x4+1))), 0, 1/(x5* ...;
(1/x4 -1/(x4+1))),0];

G=[R*x5/((4*x2 + 28 + x4*18)*Vc)*(u2 + u3 - u1 +delh20 ); ...;
(1 + x4)*R*x5/((4*x2 + 28 + 18*x4)*Vc*x1)*(u2 - x2* ...;
u1); (1 +x4)*R*x5/((4*x2 + 28 + 18*x4)*Vc*x1*Q)*delchx;...;
(1 + x4)*R*x5/((4*x2 + 28 + 18.02*x4)*x1*x4*Vc*(1/x4...;
+ 1/(x4+ 1)))*u5/(1+x4)*delw];

H=[R*x5/((4*x2 +28+ 18*x4)*Vc)*(-O2 + H2O + mleak); R*x5*(1 + x4) ...;
/((4*x2 +28+ 18*x4)*Vc*x1)*(x1*mleak - O2); R*(1 + x4)*x5/(( ...;
4*x2 +28+18*x4)*Vc*x1*Q)*(qh + qm + H2O*vap); R*x5*(1 + x4)/(( ...;
4*x2 +28 +18*x4)*Vc*x1*x4*(1/x4- 1/(x4 + 1)))*H2O];

%     The following sequence will calculate the change in x over the
%      integration time period

xdot=inv(eye(4)-F)*(G + H);
x(:,i+1)=x(:,i) + xdot*T;
u(:,i+1)=u(:,i);
```

```
end
%*********************************************************************
%    The following code will calculate the variables utilized in the recursive least
%    squares algorithm and will vary the steady state control inputs for the first 300
%    integration time steps in order for the parameter matrix to converge

if i<300
z=x(:,i+1)-xset;
v=u(:,i)-uset;
phi=[z',v']';
zdot(:,i)=xdot;

%    The following code calls the recursive least squares subroutine for
[th1,q1]=rls(th1,q1,phi,zdot(1,i));
[th2,q2]=rls(th2,q2,phi,zdot(2,i));
[th3,q3]=rls(th3,q3,phi,zdot(3,i));
[th4,q4]=rls(th4,q4,phi,zdot(4,i));

%    w(:,1) represents the diffference between the estimated states and the actual
%    output signals

w(:,i)=(phi'*[th1,th2,th3,th4])'-xdot;

%    Random variation in steady state inputs

u(1,i+1)=u(1)+0.05*rand;
u(2,i+1)=u(2)+0.05*rand;
u(3,i+1)=u(3)+0.05*rand;
u(4,i+1)=u(4)+0.01*rand;
u(5,i+1)=u(5)+5*rand;

end

%    The controller is turned on at integration time step greater than 300.  The
%    recursive least squares parameter estimator is left on line for contimued
%    system identification.

if i>300
z=x(:,i+1)-xset;
v=u(:,i)-uset;
phi=[z',v']';
zdot=xdot;
[th1,q1]=rls(th1,q1,phi,zdot(1));
[th2,q2]=rls(th2,q2,phi,zdot(2));
[th3,q3]=rls(th3,q3,phi,zdot(3));
[th4,q4]=rls(th4,q4,phi,zdot(4));
a=[th1(1:4)';
  th2(1:4)';
  th3(1:4)';
  th4(1:4)'];
b=[th1(5:9)';
  th2(5:9)';
  th3(5:9)';
  th4(5:9)'];

%    Linear quadratic Regulator implementation
%    The q vector is for the states  relative weighting and c vector for the input
%    relative weighting
q=3*[7,0,0,0;0,1800,0,0;0,0,10,0;0,0,0,40];
c=[.01,0,0,0,0;0,.01,0,0,0;0,0,.1,0,0;0,0,0,0.01,0;0,0,0,0,0.10];
```

```
m=lqr(a,b,q,c);

%    Control law implementation where m is the gain matrix
u(:,i+1)=-m*z;

u(:,i+1)=u(:,1)+u(:,i+1);

%    Constraints on calculated inputs

if u(1,i+1)<0
  u(1,i+1)=0.0;
  end

if u(2,i+1)<0
  u(2,i+1)=0.0;
  end

if u(3,i+1)<0
  u(3,i+1)=0.0;
  end

if u(4,i+1)<0
  u(4,i+1)=0.0;
  end

if u(5,i+1)<0
  u(5,i+1)=0.0;
  end

if u(1,i+1)>umax(1)
    u(1,i+1)=umax(1);
    end

if u(2,i+1)>umax(2)
    u(2,i+1)=umax(2);
    end

if u(3,i+1)>umax(3)
    u(3,i+1)=umax(3);
    end

if u(4,i+1)>umax(4)
    u(4,i+1)=umax(4);
    end

if u(5,i+1)>umax(5)
    u(5,i+1)=umax(5);
    end

end;
end;

%    Plots of states versus time

plot(time,x(1,:));
xlabel('Time (hours)')
ylabel('Pressure (kPa)')
title('Chamber Pressure')
meta bhm1;
pause;
```

```
plot(time,x(2,:));
xlabel('Time (hours)')
ylabel('Mass Fraction of O2 (Dry Air)')
title('Mass Fraction of O2')
meta bhm2;
pause;

plot(time,x(3,:));
xlabel('Time (hours)')
ylabel('Temperature (C)')
title ('Chamber Temperature')
meta bhm3;
pause;



plot(time,x(4,:));
xlabel('Time (hours)')
ylabel('Absolute Humidity (kg H2O/kg dry air)')
title('Chamber Humidity')
meta bhm4;
pause;




%    Plots of inputs versus time

xlabel('Time (hours)')
ylabel('Vacuum Flow Rate (kg/hr)')
title('Vacuum Flow Rate')
pause;

plot(time,u(2,:))
xlabel('Time (hours)')
ylabel('Oxygen Flow Rate (kg/hr)')
title('Oxygen Flow Rate')
pause;

plot(time,u(3,:))
xlabel('Time (hours)')
ylabel('Nitrogen Flow Rate (kg/hr)')
title('Nitrogen Flow Rate')
pause;


plot(time,u(4,:))
xlabel('Time (hours)')
ylabel('Temperature (C)')
title('CHX Entrance Temperature')
pause;

plot(time,u(5,:))
xlabel('Time (hours)')
ylabel('Air Flow (kg/hr)')
title('CHX Air Flow Rate')
pause;
```

```
**************************************************************************
function [thnew, Qnew]=rls(thold, Qold, phi, zdot)
%   function [thnew, Qnew]=rls(thold, Qold, phi, zdot)
% Recursive Least Squares for estimating the rows of the A and B matrices.
% th= parameter vector (in our case 9 by 1)
% Q = positive definite matrix (9 by 9). Initialize it at Q=alpha*eye(9),
%                 with alpha a small positive number (sey 0.0001)
% phi=9 by 1 vector = [z1(t),....z4(t),v1(t),....v5(t)]'
Qnew=Qold+phi*phi';
P=inv(Qnew);
thnew=thold+P*phi*(zdot-phi'*thold);
end   % rls

**************************************************************************
```

2. The second section presents the model and Fortran computer code utilized in the development of the direct adaptive control routine [Ref. 13].

a. Modified Matlab Model

```
%*****************************************************************

%                                        Program cerc.m

%*****************************************************************
%   This program is a simulation of the temperature, composition,pressureand
%   humidityof a Controlled Environment Research Chamber that is currently
%   undergoing renovation at NASA Ames Research Center located at Moffet Field,
%   CA


% The variables utilized for state definition and control are as follows:

%    x(1) = Chamber pressure (kPa)
%    x(2) = Mass fraction of O2 in dry air
%    x(3) = Chamber temperature (degrees C)
%    x(4) = Chamber absolute humidity (kg H2O/kg dry air)
%    u(1) = Vacuum flow rate out of chamber (kg/hr)
%    u(2) = Mass flow rate of oxygen into chamber (kg/hr)
%    u(3) = Mass flow rate of nitrogen into chamber (kg/hr)
%    u(4) = Coolant stream entrance temperature into CHX (degrees C)
%    u(5) = Air flow rate from chamber into CHX (kg/hr)

% The following are variables that are utilized in system defintion:

%    U     = Heat exhanger performance parameter (KJ/m^2-C)
%    A     = Heat exchanger area  (m^2)
%    mc    = Coolant flow rate across CHX (kg/hr)
%    R     = Universal gas constant  8.314 kJ/kgmole K
%    Vc    = Chamber volume (m^3)
%    mleak = Leakage of ambient air into the chamber (kg/hr)
%    xl    = Mass fraction of O2 in ambient (outside) air
%    O2    = Human consumption of O2 (four persons) (kg/hr)
%    H2O   = Human production of H2O (four persons) (kg/hr)
%    qh    = Human sensible heat load (four person)  (kJ/hr)
%    qm    = Machinery/miscellaneous sensible heat loads (kJ/hr)
%    vap   = Heat of vaporization of H2O from human in chamber-2287 kJ/hr
```

```
%    hvap1 = Heat of vaporization of H2O in CHX exit stream
%    hv    = Heat of vaporization of H2O ??? - 2213 kJ/hr
%    Q     = Matrix constant
%    x5    = Chamber temperature in degrees K
%    x6    = CHX air exit temperature in degrees K
%    a1,a2,a3,b1,b2,b3 = Coefficients of heat capacities for
%                        chamber constituents (O2,N2,H2O)

% The simulation of the multivariable, non linear equations will be
% set up as follows:
%
%    xdot= F(x)*xdot + G(x,u) + H(x,w);
%
% Solving for xdot ;
%
%    xdot= inv(1-F(x))*G(x,u) + inv(1-F(x))*H(x,w)

%-------------------------------------------------------------------------
%-------------------------------------------------------------------------

% Setting the initial conditions of the states and control variables

x = [34.464; 0.63420;21.1;0.023791];
u = [0.0025;0.3856;1.72e-3;1.5;1000];

%-------------------------------------------------------------------------
%-------------------------------------------------------------------------

% Setting the other variables

Vc = 157;
a1 = 0.9324;
a2 = 0.90468;
a3 = 1.831;
b1 = 0.000184;
b2 = 0.000204;
b3 = 0.000744;
qm = 8586.33;
R  = 8.314;
xl = 0.2344;
vap = 2287.0;
hv  = 2213.0;
mleak = 0.0025;

%-------------------------------------------------------------------------
%-------------------------------------------------------------------------

% Overall number of integration time steps

n = 1200;
p = 6;
r = p-1;

% T is the sampling duration required in the integration of the equations
% of motion that define the dynamics of the processes involved in
% regulating temperature, humidity, pressure and composition.

T = 0.004167;
time = [1:n*p+1];
time = time*T;
```

```
% Simulation for n integration steps

for k = 1:n;

  if k-2 < 0
    t = 0.0;
    ! delete time.dat;*
    fprintf('time.dat', '%e', t)
    ! run [finn.matlab]bruce3
  end;

%-------------------------------------------------------------------------
%-------------------------------------------------------------------------

% The simulation will run for p integration time steps prior to sampling
% and evaluating the data for parameter estimation and control input.

  for i = p*k-r:p*k

    O2 = 0.3856;
    qh = 2213;
    H2O = 0.67;
    x5 = x(3,i) + 273;
    x6 = u(4,i) + 281.33;
    x7 = u(4,i) + 8.33;

    x1 = x(1,i);
    x2 = x(2,i);
    x3 = x(3,i);
    x4 = x(4,i);
    u1 = u(1,i);
    u2 = u(2,i);
    u3 = u(3,i);
    u4 = u(4,i);
    u5 = u(5,i);

% Calculate the partial pressure of H2O in the exit stream of the
% CHX and the accompanying heat of vaporization

    Pstar = 10^(28.5905-8.2/2.306*log(x6) + 0.002484*x6- 3142/x6);

% Calculate the heat of vaporization of the moisture in the exit
% stream of the CHX

    hvap1 = (13.11 - log(Pstar*0.9869))*R/18.02*x6;

    if x3 < x7
      Pstar = 0.0;
    end;

% Calculate the humidity of the CHX exit stream

    we = 18.02/(4*x2+28)*(100*Pstar/(x1-100*Pstar));

% Calculate the enthalpies of the entrance and exit gases of the CHX

    CpT1 = x2*(a1*x3 + b1*x3^2) + (1-x2)*(a2*x3 + b2*x3^2) ...;
        + x4*(a3*x3 + b3*x3^2);
    CpT2 = x2*(a1*x7 + b1*x7^2) + (1-x2)*(a2*x7 + b2*x7^2) ...;
        + we*(a3*x7 + b3*x7^2);
```

```
% Calculate the change in sensible heat removed across the CHX

    delh = CpT2 - CpT1;

    if delh > 0;
      delh = 0.0;
    end;

% Calculate the change in absolute humidity across the CHX

    delw = we -x4;

    if delw > 0;
      delw = 0.0;
    end;

% Calculate the latent heat removed across the CHx

    dell = delw*hvap1;
    delh2o = (u5/(1+x4))*delw;

% Calculate the total heat removed across the heat excahnger

    delchx = (u5/(1+x4))*(delh+dell);

    if delchx > 0
      delchx = 0.0;
    end;

% Calculate the function Q

    Q = (x2*(a1 - a1*x3/x5 + 2*b1*x3 - b1*x3^2/x5) + (1 - x2)* ...;
       (a2 - a2*x3/x5 + 2*b2*x3 - b2*x3^2/x5) + x4*(-hv/x5 + ...;
       a3 - a3*x3/x5 + 2*b3*x3 - b3*x3^2/x5));

% The following is the mathematical representations of the matrices
% F, G, and H.

    F = [0, 0, x1/x5, 0; -x2/x1, 0, x2/x5, x2/(1 + x4); -(x2* ...;
       (a1*x3 + b1*x3^2) + (1 - x2)*(a2*x3 + b2*x3^2) + x4* ...;
       (hv + a3*x3 + b3*x3^2))/(x1*Q), - (a1*x3 + b1*x3^2 - ...;
       a2*x3 - b2*x3^2)/Q, 0, -((-1/(1 + x4))*(x2*(a1*x3 + ...;
       b1*x3^2) + (1 - x2)*(a2*x3 + b2*x3^2)) + a3*x3 + ...;
       b3*x3^2 + hv - (x4/(1 + x4))*(hv + a3*x3 + b3*x3^2))/Q; ...;
       -1/(x1*(1/x4 - 1/(x4 + 1))), 0, 1/(x5*(1/x4 -1/(x4 + 1))), 0];

    G = [R*x5/((4*x2 + 28 + x4*18)*Vc)*(u2 + u3 - u1 + delh2o ); ...;
       (1 + x4)*R*x5/((4*x2 + 28 + 18*x4)*Vc*x1)*(u2 - x2* ...;
       u1); (1 + x4)*R*x5/((4*x2 + 28 + 18*x4)*Vc*x1*Q)*delchx;...;
       (1 + x4)*R*x5/((4*x2 + 28 + 18.02*x4)*x1*x4*Vc*(1/x4...;
       + 1/(x4 + 1)))*u5/(1 + x4)*delw];


    H = [R*x5/((4*x2 + 28 + 18*x4)*Vc)*(-O2 + H2O + mleak); ...;
       R*x5*(1 + x4)/((4*x2 + 28 + 18*x4)*Vc*x1)*(xl*mleak - O2); ...;
       R*(1 + x4)*x5/((4*x2 + 28 + 18*x4)*Vc*x1*Q)*(qh + qm + ...;
       H2O*vap); R*x5*(1 + x4)/((4*x2 + 28 + 18*x4)*Vc*x1*x4*...;
       (1/x4 - 1/(x4 + 1)))*H2O];

% The following sequence will calculate the change in x over the
```

```
% integration time period

    xdot = inv(eye(4)-F)*(G + H);
    x(:,i+1) = x(:,i) + xdot*T;
    u(:,i+1) = u(:,i);

  end

%-------------------------------------------------------------------------
%-------------------------------------------------------------------------

% The following sends the current value of x into the fortran parameter
% estimation and adaptive control routine, written by Dr. Cory Finn, that has
% been adapted for this for this particular control problem.

  ! del y.dat;*
  y = x(:,i+1);
  fprintf('y.dat', '%e', y(1));
  fprintf('y.dat', '%e', y(2));
  fprintf('y.dat', '%e', y(3));
  fprintf('y.dat', '%e', y(4));

% The following sends the time step currently in effect to the Fortran program

  ! del time.dat;*
  t = i*T;
  fprintf('time.dat','%e',t);

% The output of the routine is the new control inputs that minimize the
% between the estimated next value of x and the setpoint.

  !run [finn.matlab]bruce3
  load inp.dat
  u(:,i+1) = inp;

%   u(:,i+1) = u(:,i);
%   if k > 25
%     u(:,i+1) = [0.0025;0.3856;1.72E-03;0;1200];
%   end;

end;

%-------------------------------------------------------------------------
%-------------------------------------------------------------------------

! copy [finn.matlab]br1.dat [finn.matlab]br_y.dat
! copy [finn.matlab]br2.dat [finn.matlab]br_u.dat
! copy [finn.matlab]br3.dat [finn.matlab]br_e.dat

! del [finn.matlab]br1.dat;*
! del [finn.matlab]br2.dat;*
! del [finn.matlab]br3.dat;*

%*************************************************************************
```

b. Fortran Code Listings

****

1. Main Program

```
*****************************************************************************


C-------------------------------------------------------------------
C-------------------------------------------------------------------
C
C     BRUCE2.PLANT2 EFFECTIVE 10/29/92
C
C-------------------------------------------------------------------
C-------------------------------------------------------------------

      INCLUDE '[FINN.MIMO]COMMON.MIMO'

C-------------------------------------------------------------------
C
C     INITIALIZATION OF VARIABLES
C
C-------------------------------------------------------------------

      KN = 4
      KM = 5

      INCLUDE '[FINN.PLANT]RAND.PLANT'

      OPEN(20,FILE='[FINN.MATLAB]TIME.DAT',STATUS='OLD',SHARED)
        READ(20,*) TIME
      CLOSE(20)

      IF (TIME.EQ.0.0) THEN

        ITER = 1

        OPEN(20,FILE='[FINN.MATLAB]STORE_SS2.DAT',STATUS='UNKNOWN')
          READ(20,*) ITER,((Y(I,J),J=1,HD1),I=1,KN),
     @             ((U(I,J),J=1,HD1),I=1,KM),
     @             ((THETA(I,J),J=1,HD1),I=1,KN),
     @             ((PHI(I,J),J=1,HD1),I=1,KN),
     @             (((UMATRIX(I,J,K),K=1,HD1),J=1,HD1),I=1,KN),
     @             ((DMATRIX(I,J),J=1,HD1),I=1,KN),
     @             (LAMDA(I),I=1,KN)
        CLOSE(20)

C       DO 50 J=1,HD1
C         Y(1,J) = 34.464/10.0
C         Y(2,J) = 0.64937*10.0
C         Y(3,J) = 21.1/10.0
C         Y(4,J) = 0.023279*500.0
C         U(1,J) = 0.0025*1000.0
C         U(2,J) = 0.3856*10.0
C         U(3,J) = 0.00172*1000.0
C         U(4,J) = 1.5*1.0
C         U(5,J) = 1000.0/1000.0
C 50    CONTINUE

C       DO 65 I=1,KN
C         DO 60 J=1,HD1
C           DO 58 K=1,HD1
C             UMATRIX(I,J,K) = 0.0
C 58        CONTINUE
C           DMATRIX(I,J)   = 1000000.0
C           UMATRIX(I,J,J) = 1.0
C           PHI(I,J)       = 0.0
C           THETA(I,J)     = 0.1
```

```
C 60      CONTINUE
C 65    CONTINUE

C       LAMDA(1) = 1.0
C       LAMDA(2) = 1.0
C       LAMDA(3) = 1.0
C       LAMDA(4) = 1.0

C       GOTO 190

    ELSE

      OPEN(20,FILE='[FINN.MATLAB]STORE.DAT',STATUS='UNKNOWN')
        READ(20,*) ITER,((Y(I,J),J=1,HD1),I=1,KN),
     @           ((U(I,J),J=1,HD1),I=1,KM),
     @           ((THETA(I,J),J=1,HD1),I=1,KN),
     @           ((PHI(I,J),J=1,HD1),I=1,KN),
     @           (((UMATRIX(I,J,K),K=1,HD1),J=1,HD1),I=1,KN),
     @           ((DMATRIX(I,J),J=1,HD1),I=1,KN),
     @           (LAMDA(I),I=1,KN)
      CLOSE(20)

    ENDIF

C-------------------------------------------------------------------------
C
C     SET PROCESS AND REFERENCE-MODEL PARAMETERS
C
C-------------------------------------------------------------------------

      NA(1) = 5
      MB(1) = 6
      NMOD(1) = NA(1) + KM*MB(1)

      NA(2) = 5
      MB(2) = 6
      NMOD(2) = NA(2) + KM*MB(2)

      NA(3) = 5
      MB(3) = 6
      NMOD(3) = NA(3) + KM*MB(3)

      NA(4) = 5
      MB(4) = 6
      NMOD(4) = NA(4) + KM*MB(4)

      LAMDAMIN = 0.000001
      MEMORYLENGTH = 1.0

      SAMPLETIME = 0.025

C       TINIT(1) = 1.2*NMOD(1)*SAMPLETIME
C       TINIT(2) = 1.2*NMOD(2)*SAMPLETIME
C       TINIT(3) = 1.2*NMOD(3)*SAMPLETIME
C       TINIT(4) = 1.2*NMOD(4)*SAMPLETIME

      TPRED = 6

      YHI(1)  =  103.0/10.0
      YLO(1)  =   34.0/10.0

      YHI(2)  =  0.64*10.0
      YLO(2)  =  0.22*10.0

      YHI(3)  =  29.44/10.0
      YLO(3)  =  15.55/10.0
```

```
      YHI(4)  =  .032*500.0
      YLO(4)  =  .0039752*500.0

      UHI(1)  =  10.0*1000.0
      ULO(1)  =   0.0*1000.0
      USTEP(1) =  1.0*1000.0

      UHI(2)  =  10.0*10.0
      ULO(2)  =   0.0*10.0
      USTEP(2) =  1.0*10.0

      UHI(3)  =  10.0*1000.0
      ULO(3)  =   0.0*1000.0
      USTEP(3) =  1.0*1000.0

      UHI(4)  =  22.0*1.0
      ULO(4)  =   0.0*1.0
      USTEP(4) =  2.0*1.0

      UHI(5)  =  2500.0/1000.0
      ULO(5)  =     0.0/1000.0
      USTEP(5) =  200.0/1000.0

      SET(1) =  34.464/10.0
      SET(2) =  0.64937*10.0
      SET(3) =  21.1/10.0
      SET(4) =  0.023279*500.0

      IF (TIME.GE.5.0) THEN
        SET(1) =  51.69/10.0
        SET(2) =  0.437851*10.0
        SET(3) =  21.1/10.0
        SET(4) =  0.015537*500.0
      ENDIF

C-----------------------------------------------------------------------
C
C     PLANT OUTPUT, PARAMETER ESTIMATION AND CONTROL
C
C-----------------------------------------------------------------------

  100 CONTINUE

      OPEN(20,FILE='[FINN.MATLAB]Y.DAT',STATUS='UNKNOWN',SHARED)
        READ(20,*) Y(1,1), Y(2,1), Y(3,1), Y(4,1)
      CLOSE(20)

      Y(1,1) = Y(1,1)/10.0
      Y(2,1) = Y(2,1)*10.0
      Y(3,1) = Y(3,1)/10.0
      Y(4,1) = Y(4,1)*500.0

C       IF (TIME.LE.TINIT(1)) THEN
C         U(1,1) = (0.0025+NRAND(ITER+10)*0.0010)*1000.0
C         U(2,1) = (0.3856+NRAND(ITER+20)*0.0010)*10.0
C         U(3,1) = (0.00172+NRAND(ITER+30)*0.0010)*1000.0
C         U(4,1) = (1.5+NRAND(ITER+40)*0.010)*1.0
C         U(5,1) = (1000.0+NRAND(ITER+50)*10.0)/1000.0
C       ENDIF

      DO 110 I=1,NA(1)
        PHI(1,I) = Y(1,NA(1)+2-I)
  110 CONTINUE

      DO 115 I=1,NA(2)
        PHI(2,I) = Y(2,NA(2)+2-I)
  115 CONTINUE
```

```
      DO 120 I=1,NA(3)
        PHI(3,I) = Y(3,NA(3)+2-I)
  120 CONTINUE

      DO 121 I=1,NA(4)
        PHI(4,I) = Y(4,NA(4)+2-I)
  121 CONTINUE

      DO 125 K=1,KN
        DO 122 I=1,MB(K)
          PHI(K,I+NA(K)) = U(1,MB(K)+2-I)
          PHI(K,I+NA(K)+MB(K)) = U(2,MB(K)+2-I)
          PHI(K,I+NA(K)+2*MB(K)) = U(3,MB(K)+2-I)
          PHI(K,I+NA(K)+3*MB(K)) = U(4,MB(K)+2-I)
          PHI(K,I+NA(K)+4*MB(K)) = U(5,MB(K)+2-I)
  122   CONTINUE
  125 CONTINUE

      CALL MULT(1)
      PREDERROR(1) = Y(1,1) - CC

      CALL MULT(2)
      PREDERROR(2) = Y(2,1) - CC

      CALL MULT(3)
      PREDERROR(3) = Y(3,1) - CC

      CALL MULT(4)
      PREDERROR(4) = Y(4,1) - CC

      CALL UDU(1)
      CALL UDU(2)
      CALL UDU(3)
      CALL UDU(4)

C  130 IF (TIME.LE.TINIT(1)) GOTO 140

  135 DO 138 I=1,HD3
        DO 137 J=1,HD3
          DIOPH(I,J) = 0.0
  137   CONTINUE
  138 CONTINUE

      CALL DIOPHANTINE(1,1)
      CALL DIOPHANTINE(1,2)
      CALL DIOPHANTINE(1,3)
      CALL DIOPHANTINE(1,4)
      CALL DIOPHANTINE(1,5)

      CALL DIOPHANTINE(2,1)
      CALL DIOPHANTINE(2,2)
      CALL DIOPHANTINE(2,3)
      CALL DIOPHANTINE(2,4)
      CALL DIOPHANTINE(2,5)

      CALL DIOPHANTINE(3,1)
      CALL DIOPHANTINE(3,2)
      CALL DIOPHANTINE(3,3)
      CALL DIOPHANTINE(3,4)
      CALL DIOPHANTINE(3,5)

      CALL DIOPHANTINE(4,1)
      CALL DIOPHANTINE(4,2)
      CALL DIOPHANTINE(4,3)
      CALL DIOPHANTINE(4,4)
      CALL DIOPHANTINE(4,5)
```

```
      CALL TABLAU

      U(1,1) = UNEW(1)
      U(2,1) = UNEW(2)
      U(3,1) = UNEW(3)
      U(4,1) = UNEW(4)
      U(5,1) = UNEW(5)

  140 CONTINUE

      CALL DISPLAY

      DO 150 I=1,KN
        DO 145 J=HD1,2,-1
          Y(I,J) = Y(I,J-1)
  145   CONTINUE
  150 CONTINUE

      DO 160 I=1,KM
        DO 155 J=HD1,2,-1
          U(I,J) = U(I,J-1)
  155   CONTINUE
  160 CONTINUE

      ITER = ITER + 1
      IF(ITER.GE.900) ITER = 1

  190 OPEN(20,FILE='[FINN.MATLAB]STORE.DAT',STATUS='UNKNOWN')
        WRITE(20,*) ITER,((Y(I,J),J=1,HD1),I=1,KN),
     @            ((U(I,J),J=1,HD1),I=1,KM),
     @            ((THETA(I,J),J=1,HD1),I=1,KN),
     @            ((PHI(I,J),J=1,HD1),I=1,KN),
     @            (((UMATRIX(I,J,K),K=1,HD1),J=1,HD1),I=1,KN),
     @            ((DMATRIX(I,J),J=1,HD1),I=1,KN),
     @            (LAMDA(I),I=1,KN)
      CLOSE(20)

      OPEN(20,FILE='[FINN.MATLAB]INP.DAT',STATUS='UNKNOWN',SHARED)
        WRITE(20,*) U(1,1)/1000.0
        WRITE(20,*) U(2,1)/10.0
        WRITE(20,*) U(3,1)/1000.0
        WRITE(20,*) U(4,1)/1.0
        WRITE(20,*) U(5,1)*1000.0
      CLOSE(20)


  999 END



C-------------------------------------------------------------------------
C-------------------------------------------------------------------------

      SUBROUTINE DISPLAY


C-------------------------------------------------------------------------
C-------------------------------------------------------------------------

      INCLUDE '[FINN.MIMO]COMMON.MIMO'

C-------------------------------------------------------------------------

C       PRINT1, TIME, Y(1,1)*10.0, Y(2,1)/10.0, Y(3,1)*10.0,
C     @         Y(4,1)/500.0
C       PRINT2, U(1,1)/1000.0, U(2,1)/10.0, U(3,1)/1000.0, U(4,1)/1.0,
```

```
C     @         U(5,1)*1000.0
C       PRINT3, PREDERROR(1), PREDERROR(2), PREDERROR(3), PREDERROR(4)

    1  FORMAT(' ', 4F12.4, F12.6)
    2  FORMAT(' ', 5F12.4)
    3  FORMAT(' ', 4F12.4)

       OPEN(20,FILE='[FINN.MATLAB]BR1.DAT',STATUS='UNKNOWN',
     @         ACCESS='APPEND')
       OPEN(21,FILE='[FINN.MATLAB]BR2.DAT',STATUS='UNKNOWN',
     @         ACCESS='APPEND')
       OPEN(22,FILE='[FINN.MATLAB]BR3.DAT',STATUS='UNKNOWN',
     @         ACCESS='APPEND')

       WRITE(20,10) TIME, Y(1,1)*10.0, Y(2,1)/10.0, Y(3,1)*10.0,
     @         Y(4,1)/500.0
       WRITE(21,11) TIME, U(1,1)/1000.0, U(2,1)/10.0, U(3,1)/1000.0,
     @         U(4,1)/1.0, U(5,1)*1000.0
       WRITE(22,13) TIME, PREDERROR(1), PREDERROR(2), PREDERROR(3),
     @         PREDERROR(4)

   10  FORMAT(' ', 4F12.4,F12.6)
   11  FORMAT(' ', 4F10.4,F14.4)
   13  FORMAT(' ', 5E14.4)

       CLOSE(20)
       CLOSE(21)
       CLOSE(22)

       RETURN
       END

C-------------------------------------------------------------------------
C-------------------------------------------------------------------------

       INCLUDE '[FINN.MIMO]MULT.MIMO'

       INCLUDE '[FINN.MIMO]UDU.MIMO'

       INCLUDE '[FINN.MIMO]DIOPH_CPLD.MIMO'

       INCLUDE '[FINN.MIMO]TABL_BRUCE2.MIMO'

       INCLUDE '[FINN.MIMO]SIMPLEX.MIMO'

C-------------------------------------------------------------------------
C-------------------------------------------------------------------------


*************************************************************************
         2.Subroutines
*************************************************************************

C-------------------------------------------------------------------------
C-------------------------------------------------------------------------
C
C      MULT.MIMO2 EFFECTIVE 8/17/89
C
C-------------------------------------------------------------------------
C-------------------------------------------------------------------------


       SUBROUTINE MULT(KK)


C-------------------------------------------------------------------------
```

```
C-------------------------------------------------------------------------

      INCLUDE '[FINN.MIMO]COMMON.MIMO'


      INTEGER KK

C-------------------------------------------------------------------------

      CC = 0.0

      DO 350 I=1,NMOD(KK)
        CC = CC + PHI(KK,I)*THETA(KK,I)
  350 CONTINUE

      RETURN
      END



C-------------------------------------------------------------------------
C-------------------------------------------------------------------------


**************************************************************************



C-------------------------------------------------------------------------
C-------------------------------------------------------------------------
C
C     UDU.MIMO2 EFFECTIVE 8/17/89
C
C-------------------------------------------------------------------------
C-------------------------------------------------------------------------


      SUBROUTINE UDU(KK)


C-------------------------------------------------------------------------
C-------------------------------------------------------------------------

      INCLUDE '[FINN.MIMO]COMMON.MIMO'


      DOUBLE PRECISION F(HD1), G(HD1), BETA(HD1)
      DOUBLE PRECISION NU(HD1), MU(HD1), LGAIN(HD1), ENTRY

      INTEGER KK, ROW, COLUMN

C-------------------------------------------------------------------------

      DO 405 ROW=1,NMOD(KK)
        ENTRY = 0.0
        DO 400 COLUMN=1,NMOD(KK)
          ENTRY = ENTRY + UMATRIX(KK,COLUMN,ROW)*PHI(KK,COLUMN)
  400   CONTINUE
        F(ROW) = ENTRY
  405 CONTINUE

      DO 410 ROW=1,NMOD(KK)
        G(ROW) = DMATRIX(KK,ROW)*F(ROW)
```

```
410  CONTINUE

     TRD = 0.0
     BETA(1) = 1.0

     DO 415 J=1,NMOD(KK)
       BETA(J+1) = BETA(J) + F(J)*G(J)
       TRD = TRD + DMATRIX(KK,J)
415  CONTINUE

     MEMORYLENGTH = TRD
     LAMDA(KK) = MEMORYLENGTH/(MEMORYLENGTH+(PREDERROR(KK)**2
    @         /BETA(NMOD(KK)+1)))
     IF(LAMDA(KK).GT.1.0) LAMDA(KK) = 1.0
     IF(LAMDA(KK).LT.LAMDAMIN) LAMDA(KK) = LAMDAMIN

     BETA(1) = LAMDA(KK)

     DO 425 J=1,NMOD(KK)
       BETA(J+1) = BETA(J) + F(J)*G(J)
       DMATRIX(KK,J) = BETA(J)*DMATRIX(KK,J)/(BETA(J+1)*LAMDA(KK))
       NU(J) = G(J)
       MU(J) = -F(J)/BETA(J)
       IF(J.EQ.1) GOTO 425
       DO 420 I=1,J-1
         UMATNEW(I,J) = UMATRIX(KK,I,J) + NU(I)*MU(J)
         NU(I) = NU(I) + UMATRIX(KK,I,J)*NU(J)
420    CONTINUE
425  CONTINUE

     DO 435 I=1,NMOD(KK)
       LGAIN(I) = NU(I)/BETA(NMOD(KK)+1)
       DO 430 J=1,NMOD(KK)
         UMATRIX(KK,I,J) = UMATNEW(I,J)
         IF(I.EQ.J) UMATRIX(KK,I,J) = 1.0
         IF(J.LT.I) UMATRIX(KK,I,J) = 0.0
430    CONTINUE
435  CONTINUE

     DO 440 I=1,NMOD(KK)
       THETA(KK,I) = THETA(KK,I) + LGAIN(I)*PREDERROR(KK)
440  CONTINUE


     RETURN
     END



C-------------------------------------------------------------------
C-------------------------------------------------------------------

**********************************************************************

C-------------------------------------------------------------------
C-------------------------------------------------------------------
C
C     DIOPH_CPLD.MIMO2 EFFECTIVE 8/17/89
C
C-------------------------------------------------------------------
C-------------------------------------------------------------------


     SUBROUTINE DIOPHANTINE(KY,KU)


C-------------------------------------------------------------------
```

```
C-----------------------------------------------------------------------

      INCLUDE '[FINN.MIMO]COMMON.MIMO'


      DOUBLE PRECISION TH(HD1), S(HD3,HD3), CONSTANT(HD3)
      INTEGER KY, KU, M, N, DMAX, DMIN

C-----------------------------------------------------------------------
C
C     INITIALIZE THE VARIABLES
C
C-----------------------------------------------------------------------

      M = NA(KY)
      N = MB(KY)

      DMIN = 1
      DMAX = MB(KY)

      DO 510 I=1,HD3
        DO 500 J=1,HD3
          S(I,J) = 0.0
  500   CONTINUE
        CONSTANT(I) = 0.0
  510 CONTINUE

      DO 520 I=1,M
        TH(I) = THETA(KY,I)
  520 CONTINUE

      DO 521 I=M+1,M+N
        TH(I) = THETA(KY,N*(KU-1)+I)
  521 CONTINUE

      TH(M+N+1) = 0.0

C-----------------------------------------------------------------------
C
C     CALCULATE THE ALPHA'S AND BETA'S CORRESPONDING TO
C     THE PAST OUTPUTS, PAST INPUTS, AND FUTURE INPUTS
C     FOR THE PREDICTED Y'S FROM Y(t+1) TO Y(t+T)
C
C-----------------------------------------------------------------------

      DO 530 J=1,N+M
        S(1,J) = TH(J)
  530 CONTINUE

C     S(1,(M+N+TPRED+1)) = TH(M+N+1)
      S(1,(M+N+TPRED+1)) = 0.0

      DO 580 I=2,M
        DO 550 J=1,(M+N+TPRED)
          DO 540 K=1,(I-1)
            S(I,J) = S(I,J) + TH(M-I+K+1)*S(K,J)
  540     CONTINUE
  550   CONTINUE

        DO 560 J=I,M
          S(I,J) = S(I,J) + TH(J-I+1)
  560   CONTINUE

        DO 570 J=(I+M),(I+M+N-1)
          S(I,J) = S(I,J) + TH(J-I+1)
  570   CONTINUE
  580 CONTINUE
```

```
      DO 620 I=(M+1),(TPRED+1)
        DO 600 J=1,(M+N+TPRED)
          DO 590 K=(I-M),(I-1)
            S(I,J) = S(I,J) + TH(M-I+K+1)*S(K,J)
590       CONTINUE
600     CONTINUE

        DO 610 J=(1+M),(1+M+N-1)
          S(I,J) = S(I,J) + TH(J-I+1)
610     CONTINUE
620   CONTINUE

      DO 624 I=2,M
        DO 622 J=1,(I-1)
          S(I,(M+N+TPRED+1)) = S(I,(M+N+TPRED+1))
     @       + TH(J)*S((I-J),(M+N+TPRED+1))
622     CONTINUE
        S(I,(M+N+TPRED+1))=S(I,(M+N+TPRED+1))+TH(M+N+1)
624   CONTINUE

      DO 628 I=(M+1),(TPRED+1)
        DO 626 J=1,M
          S(I,(M+N+TPRED+1)) = S(I,(M+N+TPRED+1))
     @       + TH(J)*S((I-J),(M+N+TPRED+1))
626     CONTINUE
        S(I,(M+N+TPRED+1))=S(I,(M+N+TPRED+1))+TH(M+N+1)
628   CONTINUE

C-------------------------------------------------------------------
C
C     CALCULATE CONSTANTS FOR THE CONSTRAINT EQNS
C
C-------------------------------------------------------------------

      IF (KU.GE.2) GOTO 680

      DO 650 I=1,(TPRED+1)
        DO 630 J=1,M
          CONSTANT(I) = CONSTANT(I) + S(I,J)*PHI(KY,J)
630     CONTINUE

        DO 640 J=(M+1),(M+DMAX)
          CONSTANT(I) = CONSTANT(I) + S(I,J)*PHI(KY,J)
640     CONTINUE

        CONSTANT(I) = CONSTANT(I) + S(I,(M+N+TPRED+1))
650   CONTINUE

      DO 670 I=1,TPRED
        DO 660 J=1,TPRED
          DIOPH(TPRED*(KY-1)+I,J) = S(I+1,J+M+DMAX)
660     CONTINUE
        DIOPH(TPRED*(KY-1)+I,KM*TPRED+1) = CONSTANT(I+1)
670   CONTINUE

      GOTO 730

680   CONTINUE

      DO 700 I=1,(TPRED+1)
        DO 690 J=(M+1),(M+DMAX)
          CONSTANT(I) = CONSTANT(I) + S(I,J)*PHI(KY,(KU-1)*N+J)
690     CONTINUE
700   CONTINUE

      DO 720 I=1,TPRED
```

```
      DO 710 J=1,TPRED
        DIOPH(TPRED*(KY-1)+1,J+(KU-1)*TPRED)= S(I+1,J+M+DMAX)
710    CONTINUE
      DIOPH(TPRED*(KY-1)+1,KM*TPRED+1)=
     @        DIOPH(TPRED*(KY-1)+1,KM*TPRED+1)+ CONSTANT(I+1)
720  CONTINUE

730  RETURN
     END


C-------------------------------------------------------------------
C-------------------------------------------------------------------



*******************************************************************************

C-------------------------------------------------------------------
C-------------------------------------------------------------------
C
C     TABL_BRUCE2.MIMO2 EFFECTIVE 10/29/92
C
C-------------------------------------------------------------------
C-------------------------------------------------------------------


      SUBROUTINE TABLAU


C-------------------------------------------------------------------
C-------------------------------------------------------------------

      INCLUDE '[FINN.MIMO]COMMON.MIMO'


      INTEGER KY, KU, M, N, T, TAG

C-------------------------------------------------------------------

      T = TPRED
      TAG = 1

      N = KN
      M = KM

800   IER = 0

      DO 810 I=1,HD4
        DO 805 J=1,HD5
          A(I,J) = 0.0
805     CONTINUE
        B(I) = 0
        BORIG(I) = 0
        D(I) = 0.0
        W(I) = 0.0
810   CONTINUE

      DO 815 I=1,HD5
        C(I) = 0.0
815   CONTINUE

C-------------------------------------------------------------------

      DO 830 KU=1,M
        A((KU-1)*T+1,(KU-1)*2*T+1)= 1.0
        A((KU-1)*T+1,(KU-1)*2*T+T+1)= -1.0
```

```
      C((KU-1)*2*T+1) = -USTEP(KU) + U(KU,2)
      C((KU-1)*2*T+T+1) = -USTEP(KU) - U(KU,2)
      DO 820 J=2,T
        A((KU-1)*T+J,(KU-1)*2*T+J) = 1.0
        A((KU-1)*T+J-1,(KU-1)*2*T+J) = -1.0
        A((KU-1)*T+J,(KU-1)*2*T+T+J) = -1.0
        A((KU-1)*T+J-1,(KU-1)*2*T+T+J) = 1.0
        C((KU-1)*2*T+J) = -USTEP(KU)
        C((KU-1)*2*T+T+J) = -USTEP(KU)
820   CONTINUE
      DO 825 J=1,T
        A((KU-1)*T+J,2*M*T+2*N*T+(KU-1)*2*T+J)= 1.0
        A((KU-1)*T+J,2*M*T+2*N*T+(KU-1)*2*T+T+J)= -1.0
        C(2*M*T+2*N*T+(KU-1)*2*T+J)= ULO(KU)
        C(2*M*T+2*N*T+(KU-1)*2*T+T+J)= -UHI(KU)
825   CONTINUE
830   CONTINUE

      DO 845 KY=1,N
        DO 840 J=1,T
          DO 835 I=1,M*T
            A(I,2*M*T+(KY-1)*2*T+J) = TAG*DIOPH((KY-1)*T+J,I)
            A(I,2*M*T+(KY-1)*2*T+T+J) = -TAG*DIOPH((KY-1)*T+J,I)
            A(I,4*M*T+2*N*T+(KY-1)*2*T+J)= DIOPH((KY-1)*T+J,I)
            A(I,4*M*T+2*N*T+(KY-1)*2*T+T+J)= -DIOPH((KY-1)*T+J,I)
835       CONTINUE
          C(2*M*T+(KY-1)*2*T+J) = TAG*(YLO(KY) -
     @                          DIOPH((KY-1)*T+J,M*T+1))
          C(2*M*T+(KY-1)*2*T+T+J) = TAG*(-YHI(KY) +
     @                          DIOPH((KY-1)*T+J,M*T+1))
          A(M*T+(KY-1)*2*T+J,4*M*T+2*N*T+(KY-1)*2*T+J)= 1.0
          A(M*T+(KY-1)*2*T+T+J,4*M*T+2*N*T+(KY-1)*2*T+J)= -1.0
          A(M*T+(KY-1)*2*T+J,4*M*T+2*N*T+(KY-1)*2*T+T+J)= -1.0
          A(M*T+(KY-1)*2*T+T+J,4*M*T+2*N*T+(KY-1)*2*T+T+J)= 1.0
          C(4*M*T+2*N*T+(KY-1)*2*T+J)= SET(KY) -
     @                            DIOPH((KY-1)*T+J,M*T+1)
          C(4*M*T+2*N*T+(KY-1)*2*T+T+J)= -SET(KY) +
     @                            DIOPH((KY-1)*T+J,M*T+1)
          A(M*T+(KY-1)*2*T+J,4*M*T+4*N*T+(KY-1)*2*T+J)= 1.0
          A(M*T+(KY-1)*2*T+T+J,4*M*T+4*N*T+(KY-1)*2*T+T+J)= 1.0
          C(4*M*T+4*N*T+(KY-1)*2*T+J)= 0.0
          C(4*M*T+4*N*T+(KY-1)*2*T+T+J)= 0.0
840     CONTINUE
845   CONTINUE

      DO 850 I=1,T
        D(I) = 0.000001
850   CONTINUE

      DO 851 I=(T+1),2*T
        D(I) = 0.000001
851   CONTINUE

      DO 852 I=(2*T+1),3*T
        D(I) = 0.000001
852   CONTINUE

      DO 853 I=(3*T+1),4*T
        D(I) = -0.000001
853   CONTINUE

      DO 854 I=(4*T+1),5*T
        D(I) = 0.000001
854   CONTINUE

      DO 855 I=1,2*N*T
        D(I+M*T) = 1.0
```

```
 855  CONTINUE

C-------------------------------------------------------------------------
C
C     ADD EXTRA ARTIFICIAL VARIABLES FOR PHASE I OF THE
C     SIMPLEX ROUTINE. MINIMIZE THE SUM OF THESE NEW
C     VARIABLES TO OBTAIN A BASIC FEASIBLE SOLUTION. IF
C     THE OBJECTIVE FUNCTION IS GREATER THAN ZERO, THEN
C     NO FEASIBLE SOLUTION EXISTS.
C
C-------------------------------------------------------------------------

      DO 860 I=1,(M*T+2*N*T)
         A(I,4*M*T+6*N*T+I) = 1.0
         C(4*M*T+6*N*T+I) = -1.0E03
         B(I) = 4*M*T+6*N*T+I
 860  CONTINUE

      MCON = M*T+2*N*T
      NVAR = 5*M*T+8*N*T

      CALL SIMPLEX

      DO 875 I=1,MCON
        DO 870 J=1,MCON
          IF (B(J).EQ.BORIG(I)) THEN
            IF (D(J).GT.1.0E-08) THEN
              IER = 100
              PRINT*
              PRINT865, IER
 865           FORMAT(' ','ERROR 'I6,': No feasible solutions')
            ENDIF
          ENDIF
 870    CONTINUE
 875  CONTINUE

      IF (IER.NE.0) THEN
        IF (TAG.EQ.0) THEN
          DO 878 KU=1,M
            UNEW(KU) = U(KU,2)
 878       CONTINUE
          GOTO 890
        ENDIF
        TAG = 0
        GOTO 800
      ENDIF

      DO 885 KU=1,M
        DO 880 I=1,T
          UPRED(KU,I) = W((KU-1)*T+I)
 880    CONTINUE
        UNEW(KU) = W((KU-1)*T+1)
 885  CONTINUE

      PRINT*
      PRINT*, OBJ, (UNEW(KU), KU=1,M)

 890  CONTINUE

      DO 894 I=1,HD3
        DO 892 J=1,HD3
          DIOPH(I,J) = 0.0
 892    CONTINUE
 894  CONTINUE


      RETURN
```

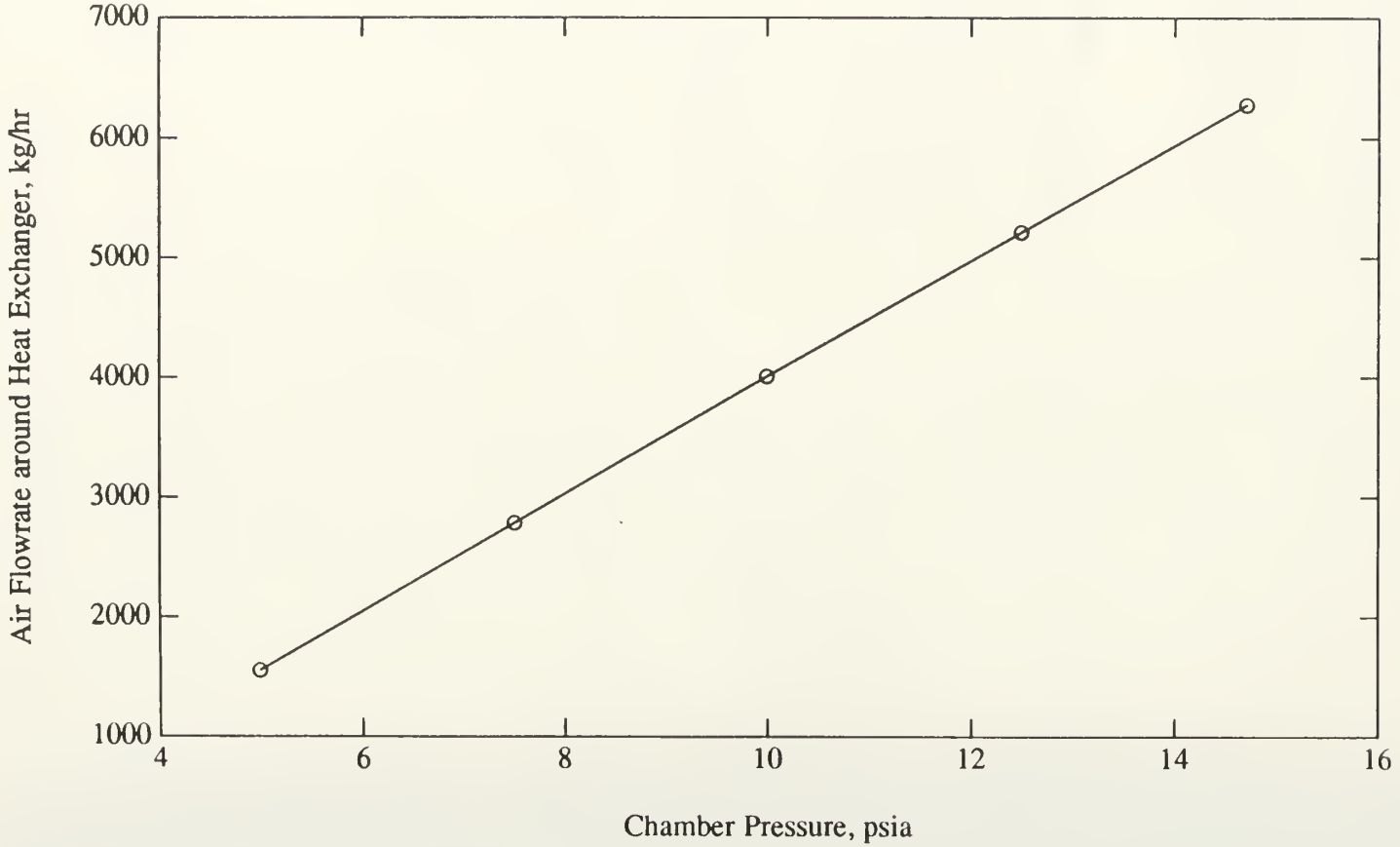
```
      END



C-------------------------------------------------------------------
C-------------------------------------------------------------------



C-------------------------------------------------------------------
C-------------------------------------------------------------------
C
C     SIMPLEX.MIMO2 EFFECTIVE 8/16/89
C
C-------------------------------------------------------------------
C-------------------------------------------------------------------


      SUBROUTINE SIMPLEX

C-------------------------------------------------------------------
C-------------------------------------------------------------------

      INCLUDE '[FINN.MIMO]COMMON.MIMO'


      DOUBLE PRECISION KMIN, RMAX, AHAT, EPS1, EPS2
      INTEGER K, R, NMULTP, NMULTD

C-------------------------------------------------------------------

      EPS1 = 1.0E-08
      EPS2 = 1.0E-04

      DO 900 I=1,MCON
        BORIG(I) = B(I)
  900 CONTINUE

      DO 910 I=1,NVAR
        A(MCON+1,I) = 0.0
        DO 905 J=1,MCON
          A(MCON+1,I) = A(MCON+1,I)+C(BORIG(J))*A(J,I)
  905   CONTINUE
        A(MCON+1,I) = A(MCON+1,I) - C(I)
  910 CONTINUE

      D(MCON+1) = 0.0
      DO 915 J=1,MCON
        D(MCON+1) = D(MCON+1)+C(BORIG(J))*D(J)
  915 CONTINUE

      NMULTD = 0
      NMULTP = 0

C-------------------------------------------------------------------

  920 KMIN = -EPS2
      K = 0
      DO 925 I=1,NVAR
        IF (A(MCON+1,I).LT.KMIN) THEN
          KMIN = A(MCON+1,I)
          K = I
        ENDIF
  925 CONTINUE

      IF (K.EQ.0) GOTO 960
```

```
      RMAX = 1.0E30
      R = 0
      DO 935 I=1,MCON
        IF ((A(I,K).GT.EPS1).AND.((D(I)/A(I,K)).LT.RMAX))
     @    THEN
          RMAX = D(I)/A(I,K)
          R = I
        ELSEIF ((A(I,K).GT.EPS1).AND.((D(I)/A(I,K)).EQ.RMAX))
     @    THEN
          DO 930 J=1,NVAR
            IF ((A(I,J)/A(I,K)).GT.(A(R,J)/A(R,K)))GOTO 935
            IF ((A(I,J)/A(I,K)).EQ.(A(R,J)/A(R,K)))GOTO 930
            RMAX = D(I)/A(I,K)
            R = I
            GOTO 935
  930       CONTINUE
        ENDIF
  935 CONTINUE

      IF (R.EQ.0) THEN
        IER = 101
        PRINT*
        PRINT940, IER
  940   FORMAT(' ','ERROR 'I6,': The solution is unbounded')
        GOTO 960
      ENDIF

      DO 950 I=1,MCON+1
        IF (I.EQ.R) GOTO 950
        AHAT = A(I,K)/A(R,K)
        IF (DABS(AHAT).LT.EPS1) AHAT = 0.0
        D(I) = D(I)-D(R)*AHAT
        IF (DABS(D(I)).LT.EPS1) D(I) = 0.0
        DO 945 J=1,NVAR
          A(I,J) = A(I,J)-(A(R,J)*AHAT)
          IF (DABS(A(I,J)).LT.EPS1) A(I,J) = 0.0
  945   CONTINUE
  950 CONTINUE

      AHAT = A(R,K)
      DO 955 J=1,NVAR
        A(R,J) = A(R,J)/AHAT
        IF (DABS(A(R,J)).LT.EPS1) A(R,J) = 0.0
  955 CONTINUE
      D(R) = D(R)/AHAT
      IF (DABS(D(R)).LT.EPS1) D(R) = 0.0

      B(R) = K

      GOTO 920

  960 CONTINUE

C-----------------------------------------------------------------------

      OBJ = D(MCON+1)

      DO 970 I=1,MCON
        W(I) = 0.0
        DO 965 J=1,MCON
          W(I) = W(I) + C(B(J))*A(J,BORIG(I))
  965   CONTINUE
  970 CONTINUE

C-----------------------------------------------------------------------

C     DO 980 I=1,MCON
```

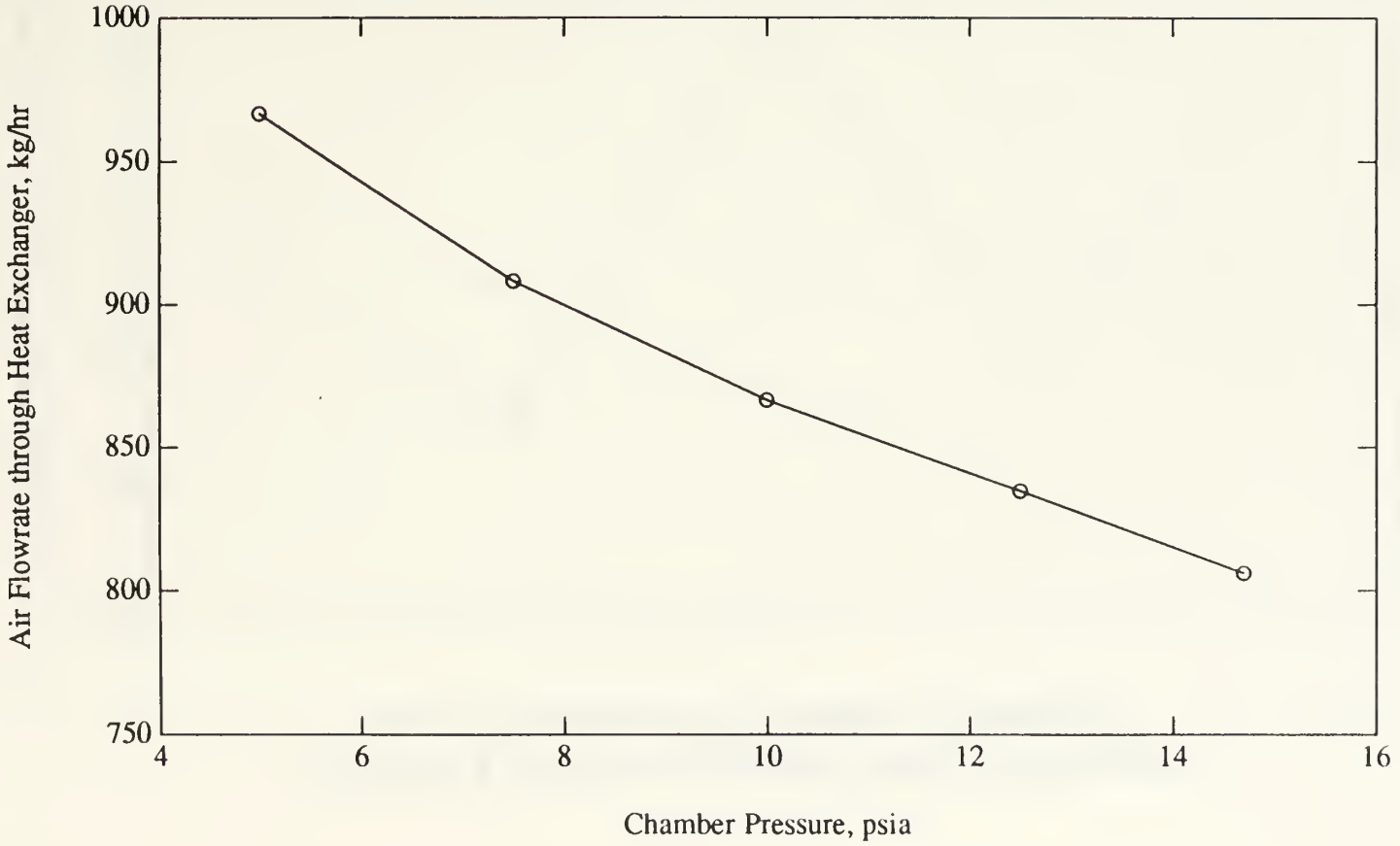
```
C         IF (DABS(D(I)).GT.EPS1) GOTO 980
C         DO 978 K=1,NVAR
C           DO 976 J=1,MCON
C             IF (K.EQ.B(J)) GOTO 978
C             IF (K.EQ.BORIG(J)) GOTO 978
C 976       CONTINUE
C           IF (DABS(A(I,K)).LT.EPS1) GOTO 978
C           AHAT = A(MCON+1,K)/A(I,K)
C           DO 974 J=1,NVAR
C             IF ((A(MCON+1,J)-(AHAT*A(I,J))).LT.-EPS1)
C     @         GOTO 978
C 974       CONTINUE
C           NMULTD = NMULTD + 1
C           PRINT*
C           PRINT972, NMULTD, I, B(I), K
C 972       FORMAT(' ','Multiple Dual Solutions',4I4)
C 978     CONTINUE
C 980   CONTINUE
C
C       DO 990 I=1,NVAR
C         DO 985 J=1,MCON
C           IF (I.EQ.B(J)) GOTO 990
C 985     CONTINUE
C         IF (DABS(A(MCON+1,I)).GT.EPS1)GOTO 990
C         NMULTP = NMULTP + 1
C         PRINT*
C         PRINT988, NMULTP, I
C 988     FORMAT(' ','Multiple Primal Solutions',2I4)
C 990   CONTINUE

C-------------------------------------------------------------------------

      RETURN

      END

C-------------------------------------------------------------------------
C-------------------------------------------------------------------------
```

APPENDIX B. PARAMETRIC STUDIES

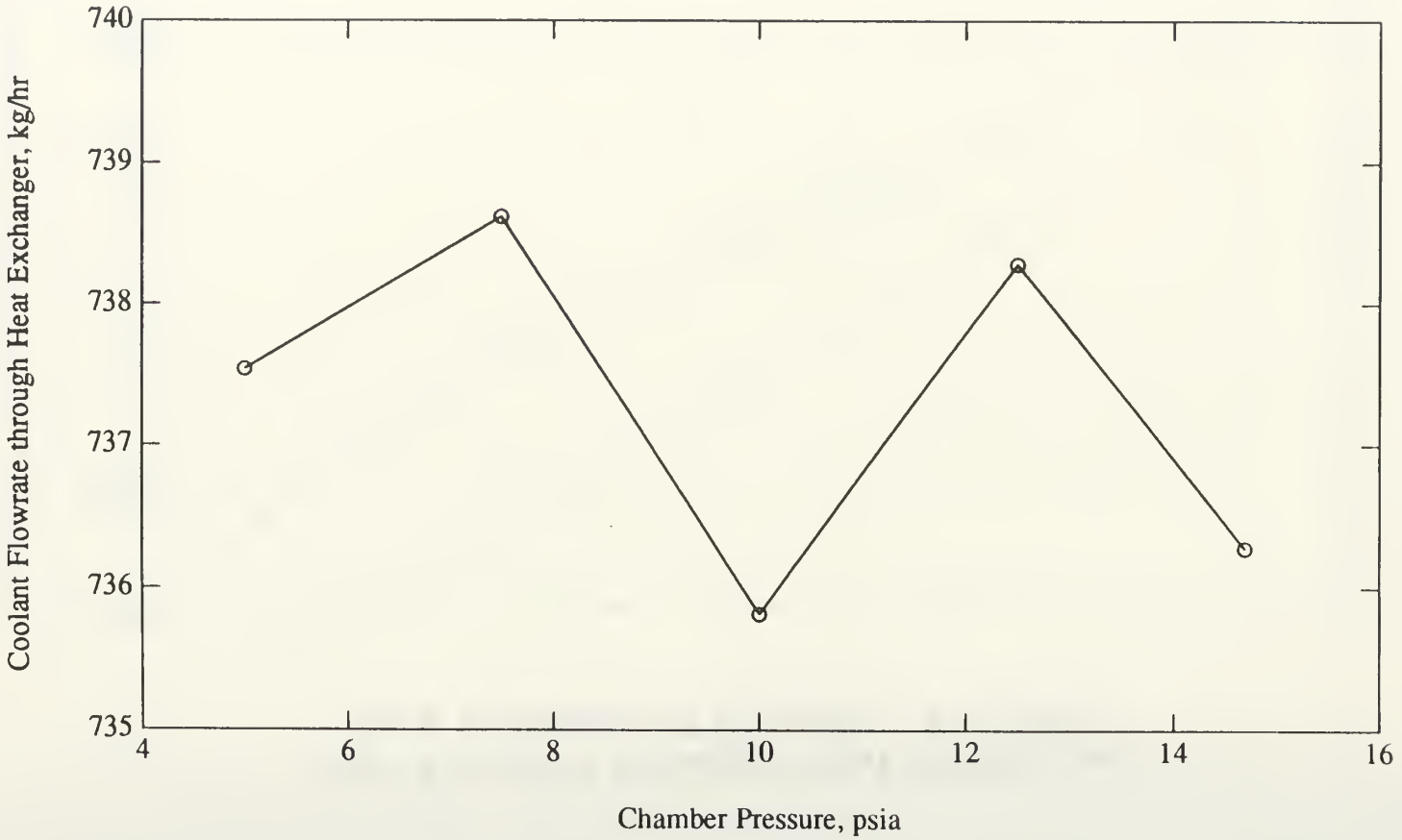
Air Flowrate around Heat Exchanger as a Function of Chamber Pressure



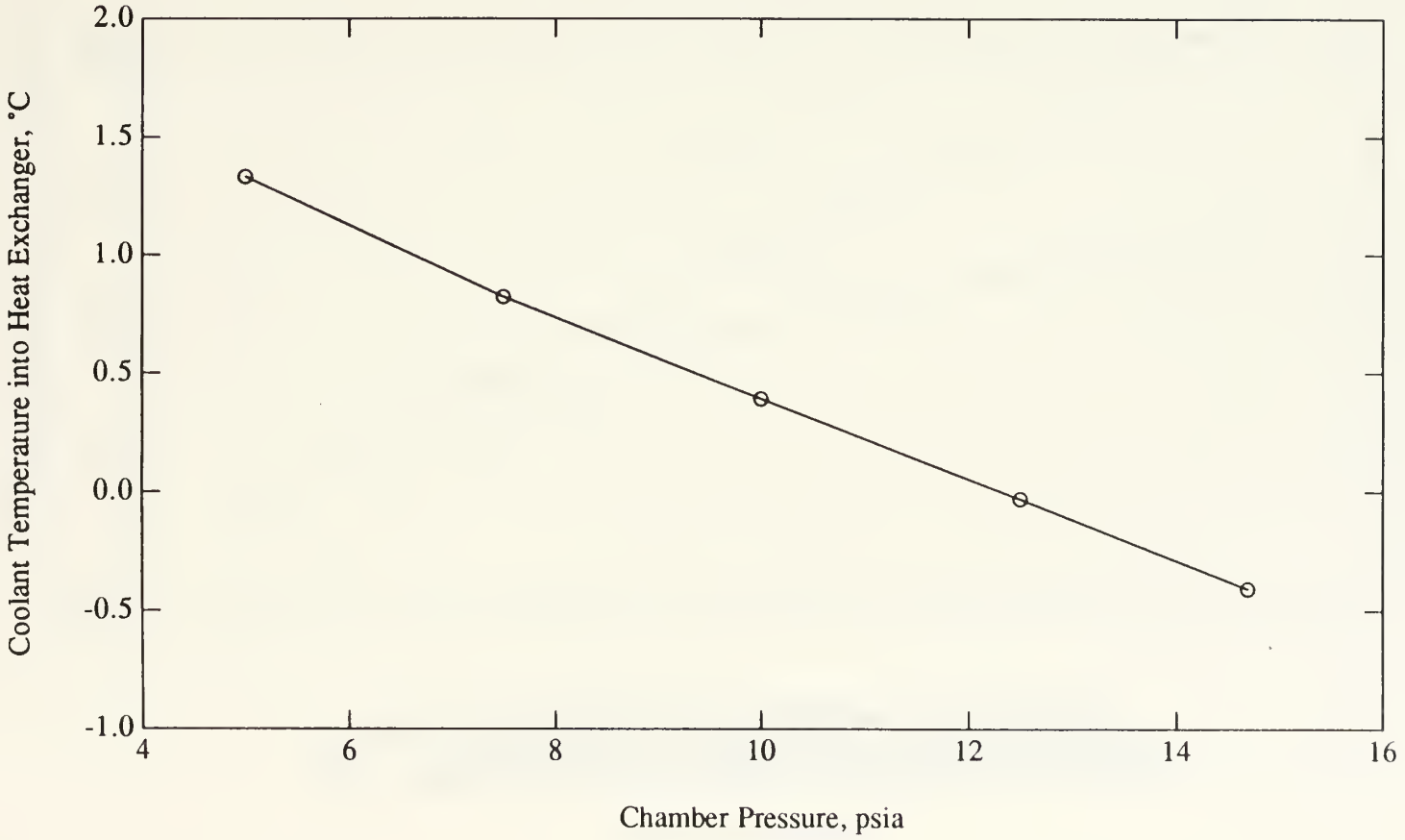
Air Flowrate through Heat Exchanger as a Function of Chamber Pressure



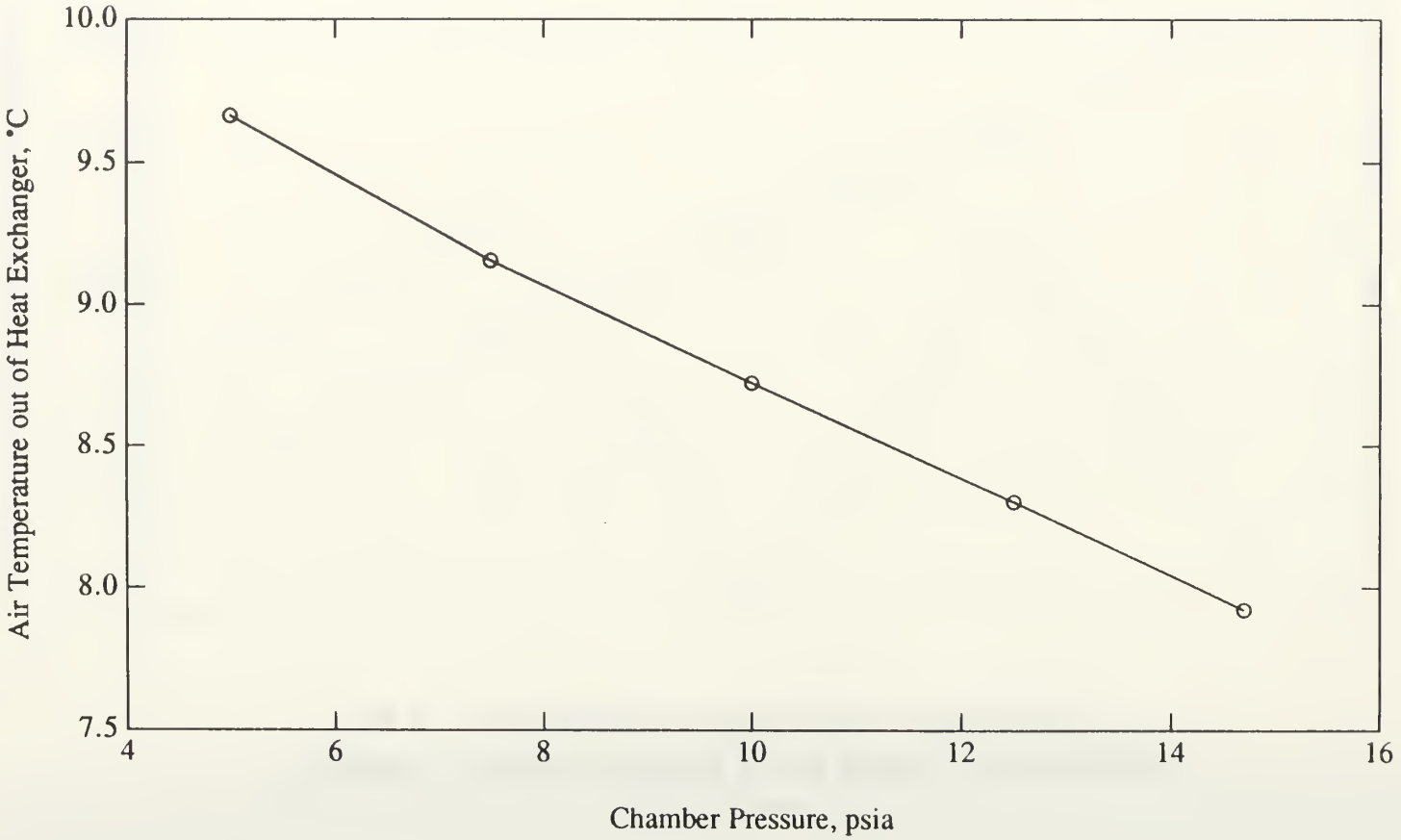
Coolant Flowrate through Heat Exchanger as a Function of Chamber Pressure



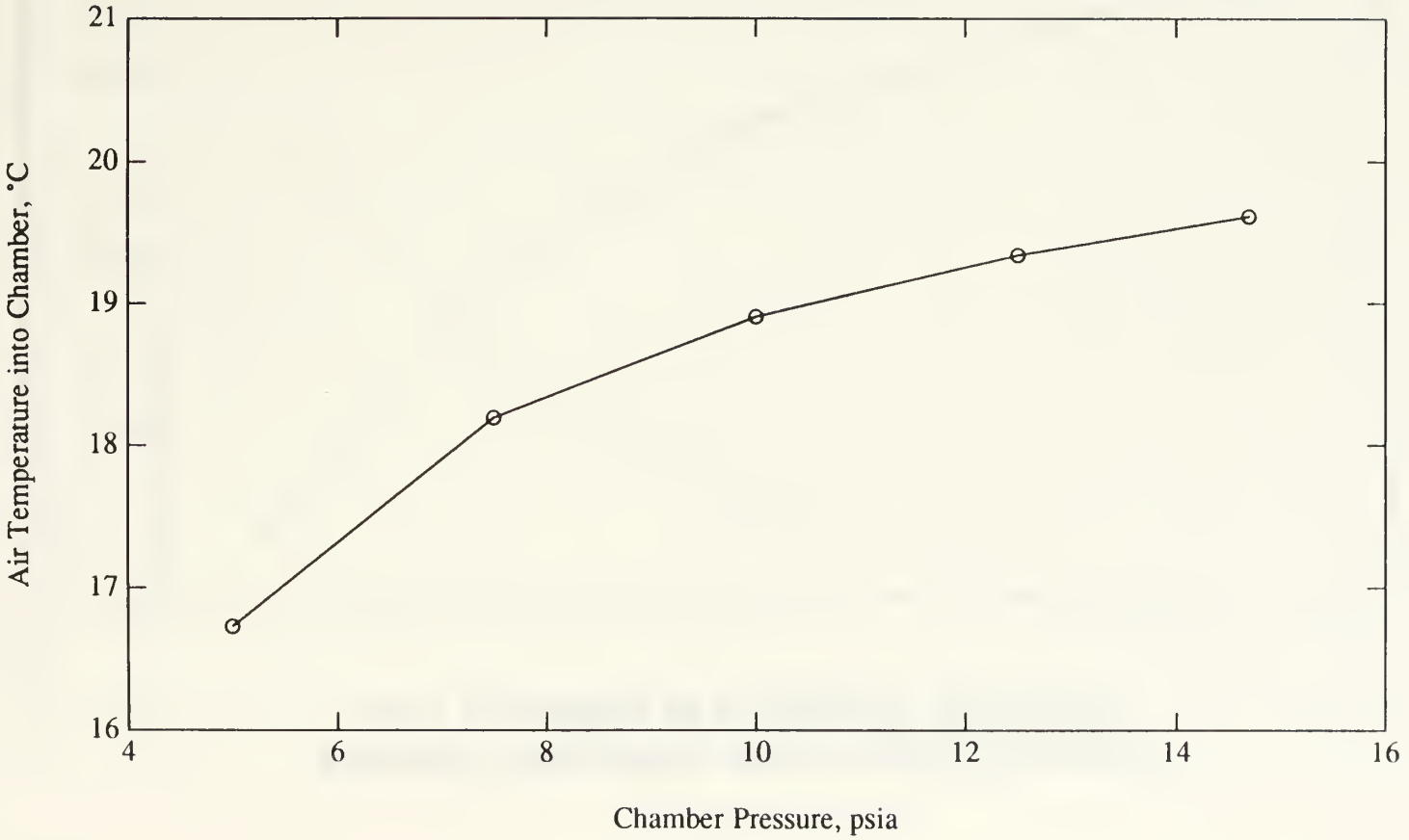
Coolant Temperature into Heat Exchanger as a Function of Chamber Pressure



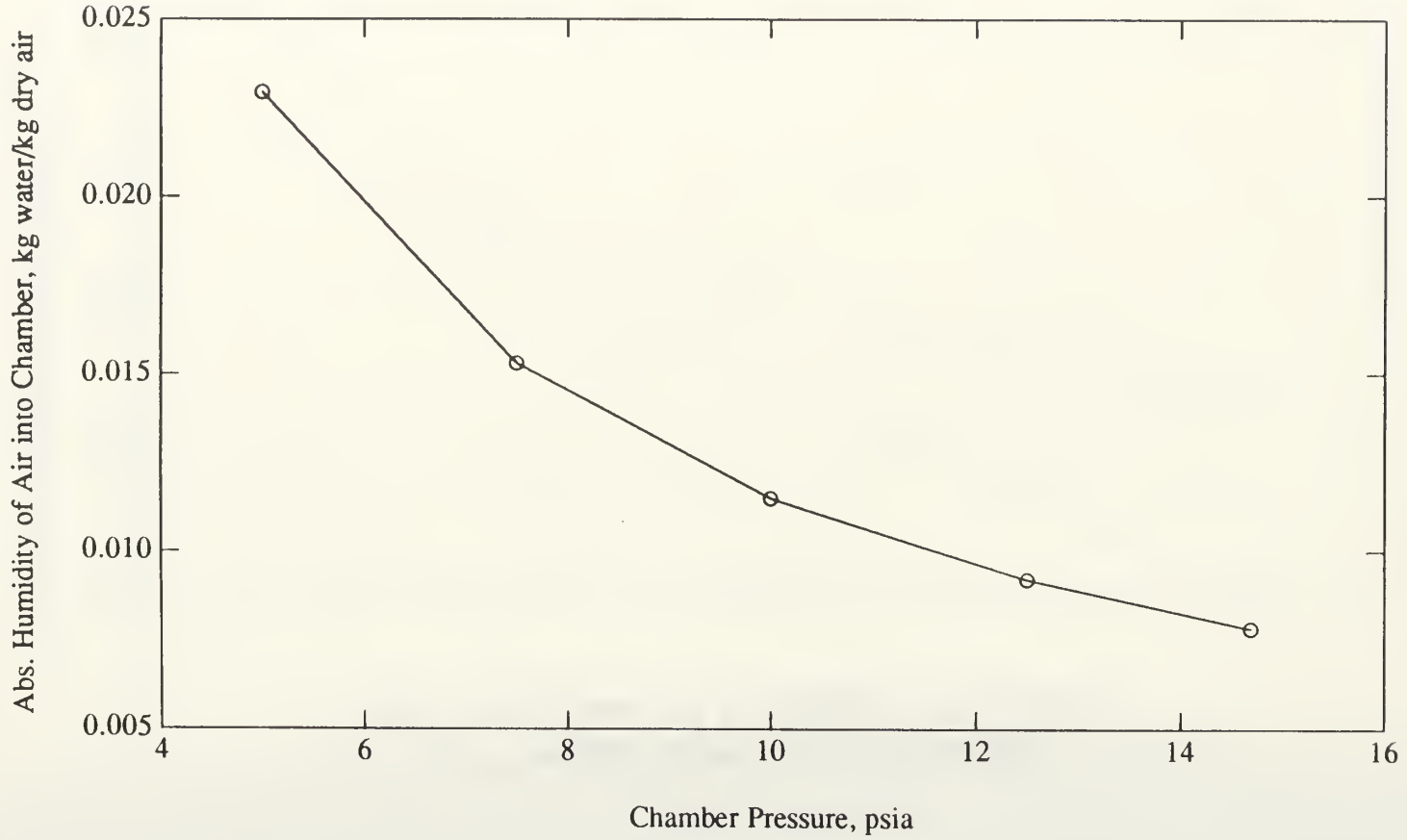
Air Temperature out of Heat Exchanger as a Function of Chamber Pressure



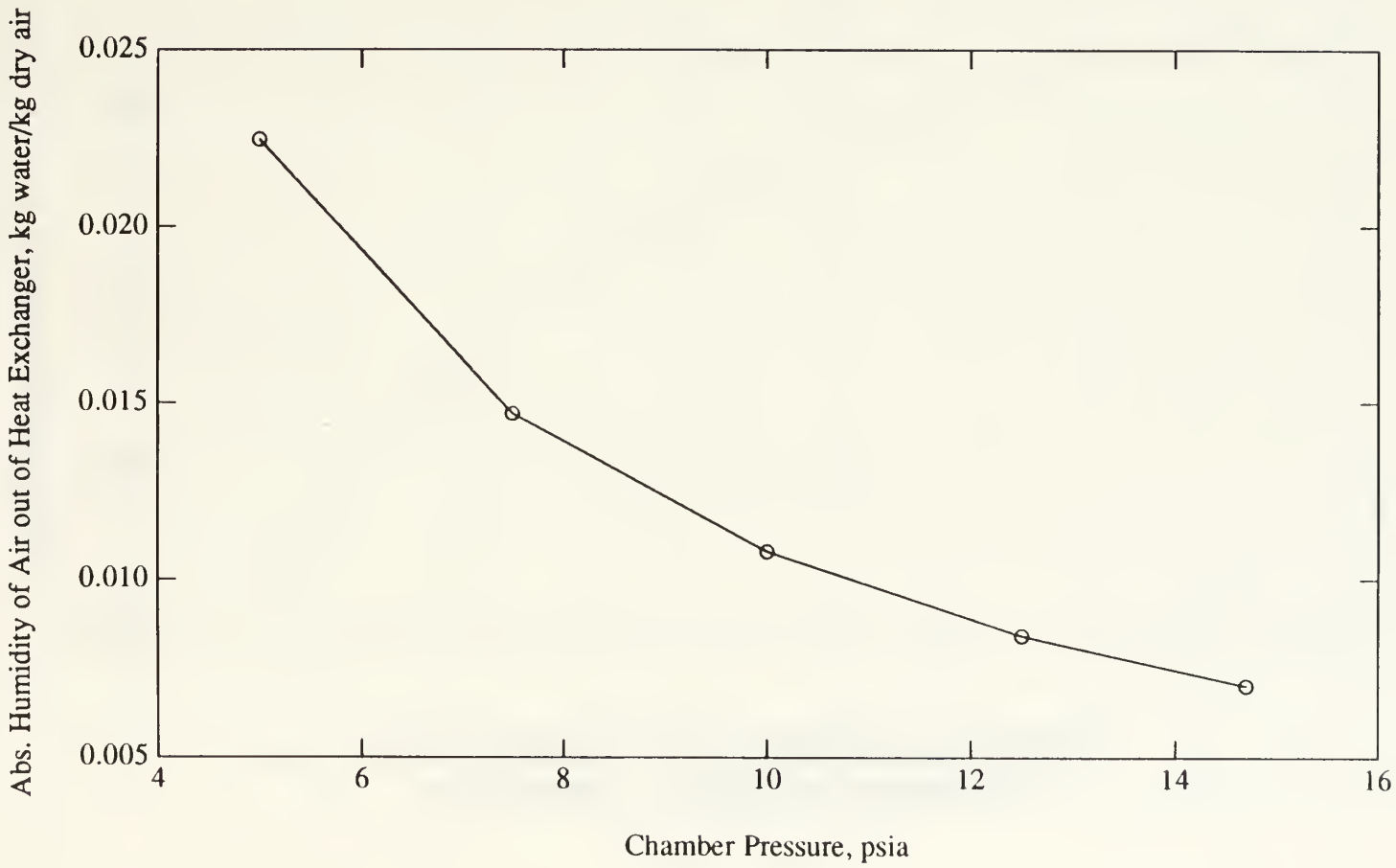
Air Temperature into Chamber as a Function of Chamber Pressure



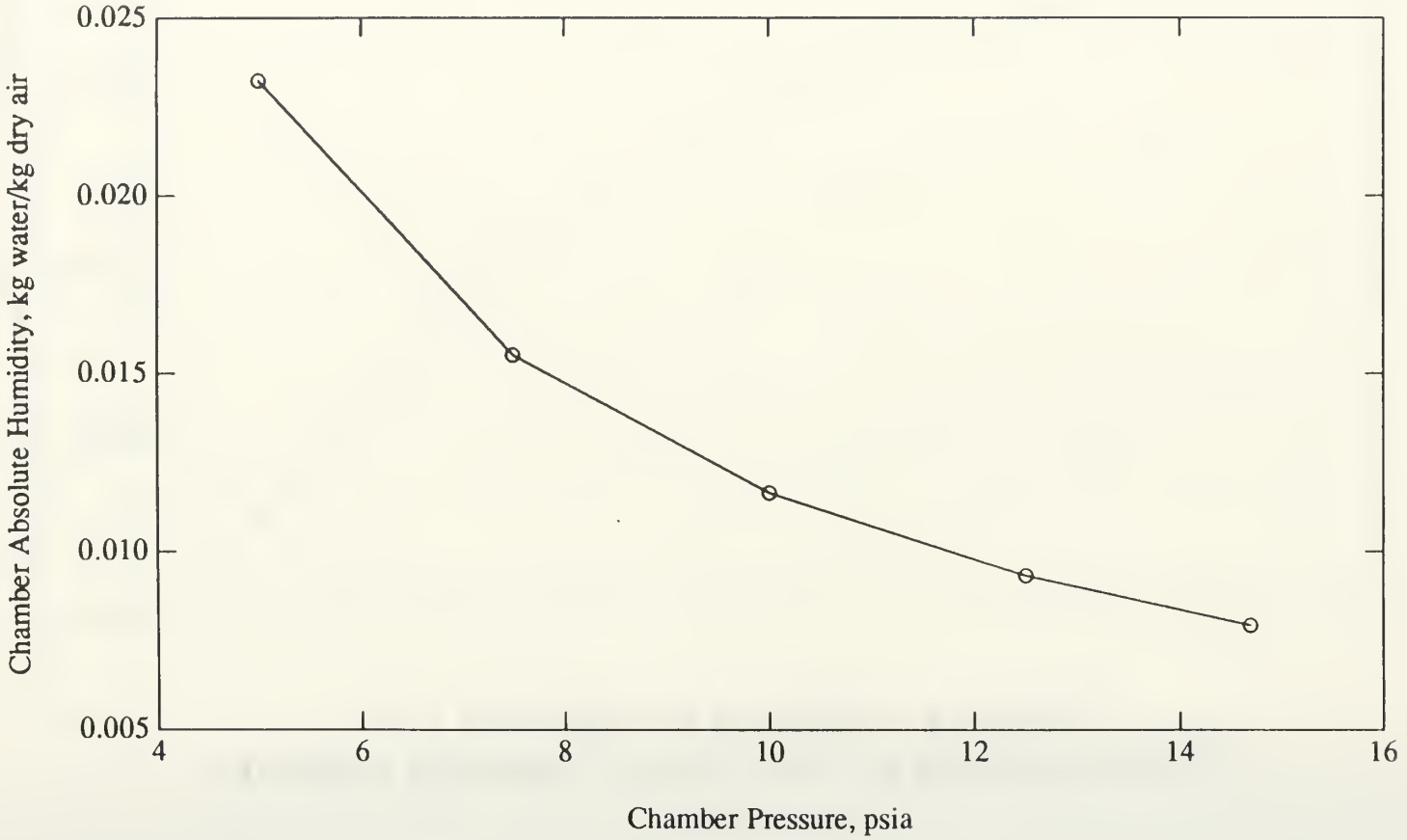
Absolute Humidity of Air into Chamber as a Function of Chamber Pressure



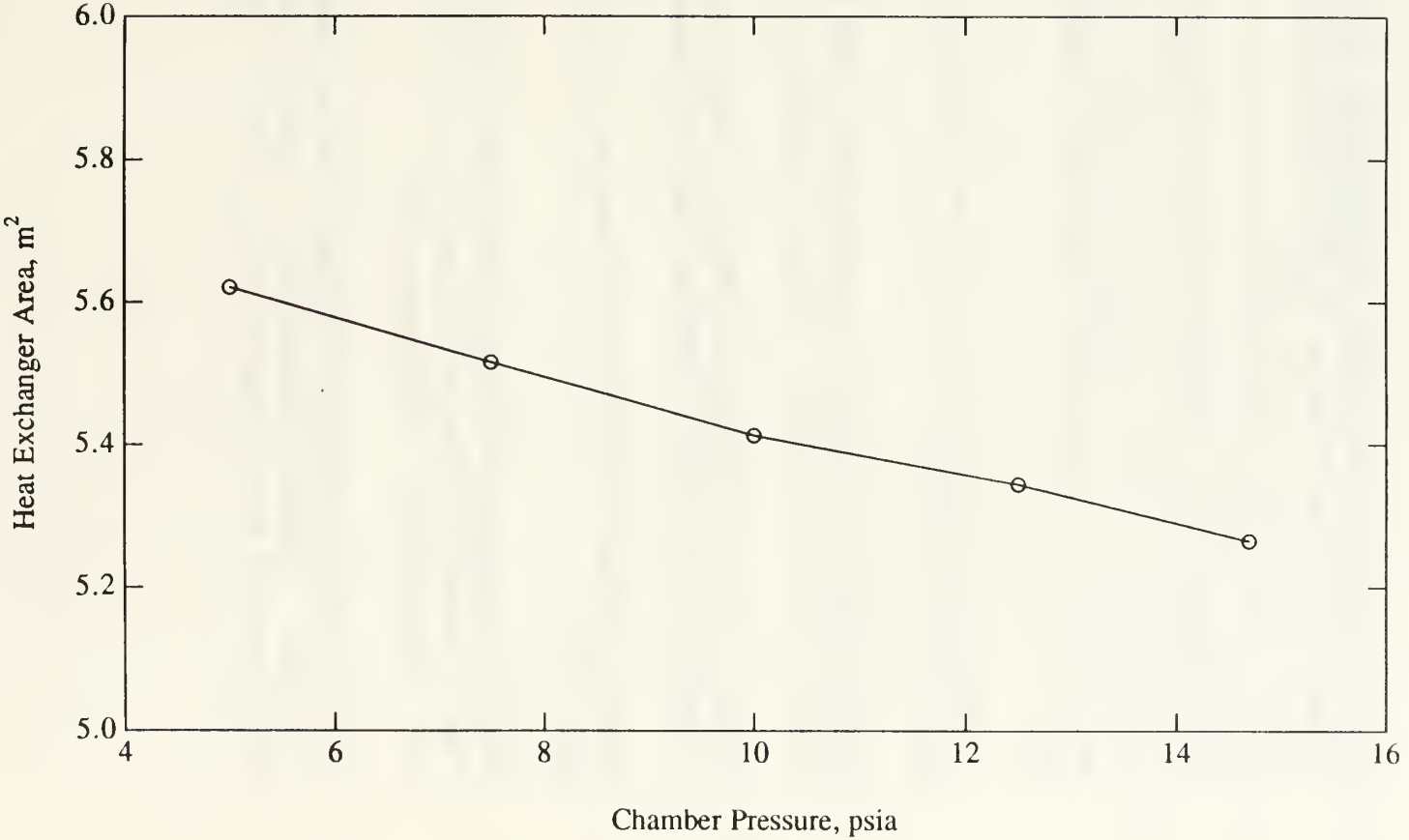
Absolute Humidity of Air out of Heat Exchanger as a Function of Chamber Pressure



Chamber Absolute Humidity as a Function of Chamber Pressure



Heat Exchanger Area as a Function of Chamber Pressure



LIST OF REFERENCES

1. National Aeronautics and Space Administration, Ames Research Center, Advanced Life Support Division, "Human Exploration Demonstration Project (HEDP) Project Plan", March, 1991.

2. Himmelblaugh, D.M., *"Principles in Chemical Engineering"*, Prentice Hall, Inc., 1974.

3. Jones, W.P., "*Air Conditioning Engineering"*, Edward Arnold Limited, 1985.

4. Telephone conversation between Dr. Richard Vann, Duke University, and the author, 05 August, 1992.

5. National Aeronautics and Space Administration, *'Man Systems Integration Standards (NASA STD-3000)"*, Volume 1, Revision A, 1986.

6. Waligora, J.W. "EVA Prebreathe Protocol", Notes from seminar presented at Lyndon B. Space Center, Houston. Texas, 09 November, 1990.

7. Ohmeda Company, Ohmeda Rascal II Anesthetic Gas Monitor Product Guide, 1992.

8. National Aeronautics and Space Administration (NASA) Ames Research Center Advanced Life Support Division, Doc. No. 90-SAS-R-005, "Initial Lunar Outpost Study" , 16 September 1991.

9. American Society of Heating, Refrigeration and Air Conditioning Engineers (ASHRAE), *"ASHRAE Fundamentals"*, American Society of Heating, Refrigeration and Air Conditioning Engineers, Inc., 1985.

10. Finn, C.K., National Aeronautics and Space Administration Ames Research Center Advanced Life Support Branch. "Controlled Environment Research Chamber Temperature and Humidity Control Parametric Studies", October 1992.

11. Astrom, K.J. and Wittenmark, B. *"Computer Controlled Systems- Theory and Design"*, Prentice Hall, 1990.

12. Kirk, D.E., *"Optimal Control Theory- An Introduction"*, Prentice Hall, 1990.

13. Finn, C.K., *"Adaptive Process Control: Modeling, Estimation, and Control of Constrained Multivariable Systems"*, Ph. D. Thesis, University of Massachussetts, Amherst, MA, 1990.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2.	Library, Code 52 Naval Postgraduate School Monterey, CA 93943-5002	2
3.	Chairman, Code AA Department of Aeronautical and Astronautical Engineering Naval Postgraduate School Monterey, CA 93943-5000	1
4.	Professor Roberto Cristi, Code EC/cx Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5000	2
5.	Professor I. Micheal Ross Code AA/Ro Department of Aeronautical and Astronautical Engineering Naval Postgraduate School Monterey, CA 93943-5000	2
6.	National Aeronautics and Space Administration Ames Research Center Systems Evaluation and Integration Branch (239-8) Advanced Life Support Division Moffet Field, CA 94035-1000	3
7.	Lt. Bruce Mathers 2739 Cypress Hill Road Carlsbad, CA 92008	3

DUDLEY KNOX LIBRARY
NAVAL POSTGRADUATE SCHOOL
MONTEREY CA 93943-5101

DUDLEY KNOX LIBRARY
3 2768 00307926 0